MICROHOBBY SEMANAL

REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES SINCLAIR

AÑO II- N.º 18

95 PTAS.
Canarias 105 ptas.

EDITA HOBBY PRESS, S.A.

PROGRAMAS

- LAS MIL CARAS
- KAMIKACES 2000
- COMECOCOS
- LAS 4 EN RAYA

BASIC

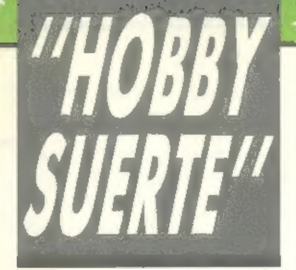
COMO TEMPORIZAR LECTURA DEL TECLADO

SOFTWARE

IN-OUT TODO SOBRE LOS "PORTS" DEL SPECTRUM

TRUCOS

¿FALLO DE LA ROM?



¡250.000 pts. en premios cada semana!

ESTE NUMERO PUEDE SIGNIFICAR UN FABULOSO REGALO PARA TI.



¡Consulta a tu Spectrum!

Cada semana, Microhobby regala 70 premios entre sus lectores. La clave del premio es el número que figura en este cupón, en la esquina superior derecha.

Para saber si el número de tu ejemplar está premiado, debes introducirlo en tu Spectrum, utilizando para ello el programa «Hobby-Suerte». La cassette con este programa se ha entregado a los lectores junto con el número 15 de Microhobby Semanal. Si no posees esta cinta, puedes pedir una copia a un amigo o por carta a **Hobby Press, S.A., Apartado n.° 54062. Madrid,** incluyendo dentro del sobre 180 pts. en 3 sellos de Correos de 60 pts. cada uno. Este programa sirve para leer todos los números durante las **50 semanas** que dure este Concurso.

(Bases en el reverso)

Premios semanales

■ Primera Categoría

Un **Spectrum 48 k.** (o un Microdrive y un Interface 1, a elegir por el interesado).

■ Segunda Categoría

Una **Impresora GP 50 de Seikosha,** especialmente diseñada para Spectrum (2 premios).

■ Tercera Categoría

Un **Joystick con su interface** (3 premios).

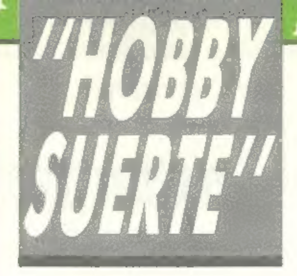
■ Cuarta Categoría

Una **Suscripción a Microhobby Semanal** por un año (50 números. Si el lector premiado ya es suscriptor, podrá optar por prolongar su suscripción anual o un premio de Quinta Categoría) (14 premios).

■ Quinta Categoría

Una **Cinta de Programa,** a elegir entre un variado surtido de juegos, utilidades, etcétera (50 premios).

Hobby Press, S.A. garantiza que cada semana introduce al azar, entre todos los ejemplares que componen la edición, **setenta cupones** correspondientes a los premios aquí citados.



Instrucciones para concursar

Una vez introducido en memoria el programa **«Hobby-Suerte»,** aparecerá en la pantalla la clásica máquina «tragaperras» de frutas.
A continuación, debes teclear el número que figura en esta tarjeta. Al pulsar «Enter», la «máquina» se pone en marcha y te hace saber si has sido agraciado con uno de los 70 premios semanales.

Muy importante: Puede ocurrir que, al introducir en el programa números al azar, alguno de ellos corresponda casualmente a un premio. También es posible, con los suficientes conocimientos de Basic, alterar el programa para que un número determinado aparezca como premiado.

Por todo ello, debemos aclarar los siguientes aspectos:

1. El único justificante para reclamar un premio determinado es la posesión del cupón con el número impreso en él.

2. Todos los números susceptibles de dar premio están registrados ante Notario.

3. Hobby Press, S.A. no se hace responsable de ningún otro cupón que no corresponda a los números previamente registrados. Tampoco se atenderán reclamaciones verbales que no vengan acompañadas por la posesión del cupón con el número premiado.

4. Cualquier lector puede solicitar de esta Editorial la comprobación de la entrega de los Premios semanales.

5. Hobby Press, S.A. se reserva el derecho a resolver según su criterio cualquier cuestión no prevista en las Bases de este Concurso.

6. La reclamación de cualquier Premio de este Concurso caduca el día 30 de Junio de 1986.

COMUNICACION DE PREMIO

(Enviar rellenado con letra clara y en sobre cerrado)

Nombre .... Edad ....
Apellidos ....
Domicilio .... Teléfono ....
Ciudad .... C.P. .... Provincia ....
Categoría del Premio Obtenido .... Número de Microhobby ....

Si consideras que tu cupón tiene premio, fotocópialo como medida de seguridad y envíalo por **Correo Certificado** a Hobby Press, S.A., Apartado 54.062 de Madrid. Por favor, anticípanos todos estos datos por teléfono, llamando al (91) 654 32 11. En este mismo número atenderemos cualquier consulta o duda sobre las Bases o la mecánica de «Hobby Suerte».

Envía este cupón por correo certificado a HOBBY PRESS. Apt. 54.062. Madrid

Director Editorial
José I. Gómez-Centurión
Director Ejecutivo
Domingo Gómez
Redactor Jefe
Africa Pérez Tolosa
Diseño
Jesús Iniesta
Maqueta
Rosa María Capitel
Redacción
José María Díaz
Gabriel Nieto
Colaboradores
Jesús Alonso, Lorenzo Cebeira, Primitivo de Francisco, Rafael Prades
Fotografía
Javier Martínez
Carlos Candel
Portada
José María Ponce
Dibujos
Manuel Berrocal, J.R. Ballesteros, A. Perera, F.L. Frontán, J. Septien, Pejo, J.M. López Moreno

Edita
HOBBY PRESS, S.A.

Presidente
María Andrino
Consejero Delegado
José I. Gómez-Centurión
Administrador General
Ernesto Marco

Jefe de Publicidad
Marisa Esteban
Secretaria de Publicidad
Concha Gutiérrez
Publicidad Barcelona
Isidro Iglesias
Tel.: (93) 307 11 13

Secretaria de Dirección
Marisa Cogorro

Suscripciones
M.ª Rosa González
M.ª del Mar Calzada

Redacción, Administración y Publicidad
La Granja, n.º 8
Polígono Industrial de Alcobendas
Telf.: 654 32 11

Dto. Circulación
Carlos Peropadre

Distribución
Coedis, S.A. Valencia, 245.
Barcelona.

Imprime
Rotedic, S.A.
Carretera de Irún, Km. 12,450
Tel.: 734 15 00
Fotocomposición
Consulgraf
Nicolás Morales, 34 - 1.º
Tel.: 471 29 08
Fotomecánica
Zescán
Nicolás Morales, 38
Tel.: 472 38 58
Depósito Legal:
M-36.598-1984

Representante para Argentina, Chile, Uruguay y Paraguay, Cia. Americana de Ediciones, S.R.L. Sud América, 1.532. Telf.: 21 24 64. 1209 BUENOS AIRES (Argentina).

MICROHOBBY no se hace necesariamente solidaria de las opiniones vertidas por sus colaboradores en los artículos firmados. Reservados todos los derechos.

Solicitado control OJD

MICROHOBBY ESTA SEMANA

Año II · N.º 18 · 5 al 11 de marzo de 1985
95 ptas. (Sobretasa Canarias 10 ptas.)


```
Buenas jefe:
Te envio en la cinta adjunta lo siguiente:

- El programa COPYUPI.
- El CONTEXT V.6 para que puedas imprimir el material que te
  mande en cassette

  Además te envio un truquito divertido para la sección
correspondiente. Por otra parte necesito que me digas si puedo
utilizar las rutinas de alta velocidad para mi próximo copiador.

                         lorenzo

P.S.- Los programas de la cinta los he sacado directamente del
      microdive, así que antes de intentarlos cargar pásalos a
      un cartucho.
```

ESTA carta que ven aquí reproducida, es una de las que, como director de la publicación, recibo diariamente de nuestros colaboradores. Sin embargo, ésta concretamente, habría de convertirse, por desgracia, en una carta especial.

Nuestro apreciado colaborador Lorenzo Cebeira, autor entre otros del programa «Microcopi», seguramente debió de echarla al correo pocas horas antes de tomar el vuelo de IBERIA que se supone habría de llevarle a Bilbao el día 19 de febrero y que, como todos ustedes ya saben, sufrió un desgraciado accidente en el que no ha habido ningún superviviente.

Con 26 años de edad, empleado desde hacía aproximadamente seis meses en la multinacional NCR, Lorenzo colaboraba con nuestra revista desde su creación y siempre le veíamos por nuestra casa con su maleta de viaje, con prisas, a caballo entre Madrid y Bilbao, donde desarrollaba la mayor parte de su labor profesional.

Lorenzo, trabajador entusiasta, me había entregado gran cantidad de material para su publicación. Todos nosotros pensamos que el mejor homenaje que podemos rendir a su recuerdo y a su trabajo, es la publicación de estos artículos. Por ello, en números próximos, nos sentiremos orgullosos de que su firma siga apareciendo en MICROHOBBY.

Desde estas líneas, escritas aún entre la sorpresa y el dolor por la muerte del amigo, queremos también expresar a los familiares de Lorenzo, nuestro pésame más sincero.

Domingo Gómez. Director

A LA PESCA DE PROGRAMAS

Todos sabemos que la afición por el software en el Reino Unido, alcanza cotas inusitadas, pero lo que ninguno podíamos imaginar es que pudieran montarse puestos más propios de una barraca de ferias, que de una Microferia.

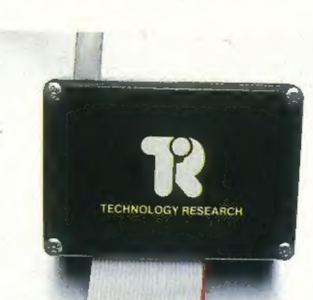
Y fue precisamente en esta última donde, ante nuestro asombro más absoluto, pudimos asistir a un espectáculo sorprendente. En uno de los stands, donde destacaban en la parte superior programas muy conocidos para el Spectrum, se encontraba un grupo de gente que, previo pago de una libra, se disponían a sacar un paquete envuelto de una caja. Parece ser que, por ese módico precio, se podía intentar pescar un programa conocido; ahora bien, con el consiguiente riesgo de que el producto pescado en cuestión, fuera otro muy distinto, lo que ocurría la mayoría de las veces. De modo que el posible «chollo» de conseguir el Knigth Lore por 200 ptas., se podía convertir, si la suerte no nos acompañaba, en más de su precio original. Eso sí, nos iríamos a casa con un montón de programas, aunque, como en este caso, fueran de dudable calidad.



INTERFACE CENTRONICS PARA EL «QL»

Ya está disponible para el QL un interface Centronics, que nos permitirá usar todas las impresoras que lleven este tipo de conexión, ya que hasta ahora, esto era uno de los inconvenientes que tenía el ordenador.

El interface ha sido creado por la casa Tecnology Research, que es la misma que hizo el Interface de Disco para el Spectrum.



Los que aducían, por tanto, que al QL le hacía falta una salida Centronics, están de enhorabuena, aunque haya tenido que ser, como en este caso, con un periférico.

ANEXO: UNIDOS CONTRA LOS PIRATAS

Tras un largo período de tiempo, en el que las empresas de Software de este país han venido sufriendo incesantemente el acoso de la piratería, se ha producido por fin una reacción: ANEXO.

ANEXO, es la Asociación Española de Empresas de Soporte Lógico, y está compuesta por 16 compañías: RCA, EDIC. SM, Publinformática, Compulogical, ALEA, ABC Analog, Music-Soft, PROEINSA, Zafiro, Dinamic, ERBE, Software Center, Idealogic e Ingelek.

La Asociación ha creado unos estatutos, que posteriormente ha modificado, y en los cuales se dictan las normas de conducta de la organización, y además, se tiene en cuenta la posible incorporación a ANEXO de otras empresas del sector.

En un principio, esta Asociación ha puesto su mirada en aquellos programas destinados a ordenadores domésticos, con el fin de proteger sus intereses en este sector, tantas veces pisoteados por los piratas de Software.

Una de las primeras medidas a tomar es informar de una manera clara, qué productos son legítimos y cuáles son piratas. Para ello, han acordado formar el Registro de Tramposos y Plagiarios, TYP. Mediante este distintivo, ANEXO informará a los profesionales del sector, sobre quiénes practican la piratería. Pretenden, de este modo, crear un organismo parecido al RAI del mundo financiero.

Por otra parte, se crea un sello de garantía al que podrán acogerse todas las empresas legales de este país. El logotipo será el siguiente: «ANEXO-HOME-ORIGINAL-AUTORIZADO».

Con todas estas medidas, se pretende crear una campaña de mentalización contra los piratas, con la ayuda, por supuesto, de los medios de prensa. Pero la batalla más importante va a ser sin duda la que tengan que librar, para que la Administración reconozca de una vez por todas, los derechos de autor de los programadores. En ese sentido, se está intentando la revisión de la ley. Ya va siendo hora.

SPIDERMAN, PARA TU SPECTRUM

Marvel Cómics y Scott Adams Internacional, han lanzado al mercado su nuevo producto, continuando con la saga de los superhéroes, que empezara con «Hulk». En esta ocasión se trata de Spiderman, el popular personaje del cómic.

El programa, al igual que el otro, es una aventura con unos gráficos muy buenos, y en el que los textos han sido simplificados al máximo, con el fin de ahorrar memoria.



SINCLAIR REDUCE SUS PRECIOS

Sorprendentemente, la compañía Sinclair ha reducido el precio de los cartuchos Microdrive en un 60%, aproximadamente, bajando de 4,95 a 1,99, con lo que el precio actual se fija, en Inglaterra, en unas 400 ptas. Esta monumental bajada de precios, ha servido ya para que se vendan al por mayor más de un millón de cartuchos.

La repercusión de esta medida puede ser realmente provechosa para el usuario, sobre todo en el caso del QL, ya que tengamos en cuenta que gracias a esta medida, los precios del software para este ordenador, pueden bajar drásticamente. De hecho, antes no se encontraban programas por menos de 20 libras (4.000 ptas.) y ahora una compañía, Quantum Software, proyecta introducir paquetes de utilidades por 6,95 libras (unas 1.300 ptas.).



En España, se espera que el precio de los cartuchos esté en torno a las 500 ptas. Mientras, se van confirmando los rumores de la bajada inmediata del Spectrum, cuyo precio oscilará alrededor de las 31.000 ptas.

LIBROS



BASIC CURSO ACELERADO

Paraninfo - Claude J. De Rossi - 224 páginas.

Como su propio nombre indica, está dirigido a personas que necesiten aprender rápidamente este lenguaje, teniendo en cuenta que el lector de este libro no tiene conocimientos sobre el tema.

Para conseguirlo, se ha escrito de una forma informal y evitando en lo posible usar complejos tecnicismos, que pudieran confundir al lector. Lo que se ha intentado en realidad, es confeccionar un texto para aprender de forma autosuficiente.

Cada uno de los capítulos, se encuentra estructurado de una forma amena y sencilla, y acompañado además de bastantes ejemplos que sirven para lograr clarificar cada uno de los temas que se explican.

Al final de cada capítulo, hay una serie de ejercicios que el lector tiene que intentar resolver, y que son resumen de todo lo explicado. Las respuestas a estos ejercicios se encuentran al final del libro.

El texto es muy claro, y está muy bien organizado, de forma que se tratan todos los comandos del Basic de una forma bastante generalizada, es decir, no limitándose al Basic del Spectrum, sino, por el contrario, ofreciéndonos una visión mucho más generalizada del lenguaje, lo que nos permite acercarnos también a otros ordenadores.

Al final de la obra, tenemos un índice alfabético de materias y comandos que nos permite localizar cualquiera de éstos, en un momento determinado.



ENCRUCIJADA

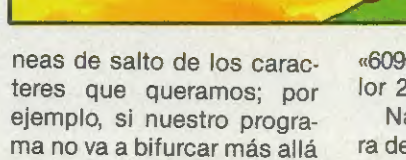
Proponemos una manera, sencilla y rápida, de acceder a distintos segmentos del programa estructurados como subrutinas mediante la elección de un número.

El método es crear una variable de cadena conteniendo TODOS los números de línea a los que queremos bifurcar cuando se cumpla una determinada condición (pulsar una tecla, cierto valor de una variable, etc...)

Luego, realizamos un cálculo de dónde se encuentra el número de línea que interesa, basándonos en la condición y en la longitud de las líneas (dada por la variable step); a continuación, fragmentamos la cadena (a$) y enviamos el control del programa a un sitio determinado mediante la función VAL.

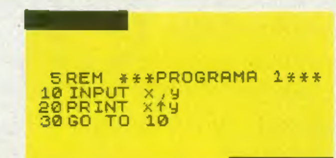
El cometido de la variable step es permitirnos utilizar líneas de salto de los caracteres que queramos; por ejemplo, si nuestro programa no va a bifurcar más allá de la línea 90, podríamos construir a$ como a$ = «6090» y asignar a step el valor 2.

Naturalmente, esta manera de enfocar el problema es igualmente válida para la sentencia GOTO.



A. PERERA

```
  10 LET a$="010002000300040005
0"
  15 LET step=4: LET sup=0: LET
inf=sup
  20 INPUT num
  25 IF num<1 OR num>LEN a$/step
 THEN GO TO 20
  30 LET sup=num*step
  32 LET inf=sup-step+(num=1)
  35 GO SUB VAL a$(inf TO sup)
  40 STOP
 100 PRINT 1: RETURN
 200 PRINT 2: RETURN
 300 PRINT 3: RETURN
 400 PRINT 4: RETURN
 500 PRINT 5: RETURN
```



```
 5 REM ***PROGRAMA 1***
10 INPUT x,y
20 PRINT x↑y
30 GO TO 10
```

Fallo de la ROM?

Todos aquellos que se interesen por las aplicaciones científico-técnicas del Spectrum, utilizándolo como una sofisticada calculadora programable, se habrán topado, en alguna ocasión, con la necesidad de calcular potencias de números o funciones.

El problema se presenta cuando queremos calcular potencias de números negativos, expresiones del tipo -3^2.

Si intentamos el cálculo en comando directo, nos encontraremos con la sorpresa de que el resultado es —9 (¡) en lugar de +9; para aumentar más aún la confusión, realizando este cálculo en modo programa, normalmente se obtiene un mensaje de error del tipo «argumento no válido», con la consiguiente detención del programa.

El programa 1 evidencia este hecho y le permitirá experimentar con los posibles valores de la base (x) y del exponente (y). El programa 2, propone una manera alternativa de solucionar el problema; en líneas generales, realiza lo siguiente:

Línea 2Ø: detecta si la base es o no menor que cero.

Línea 3Ø: si es negativo, informa de ello y asigna a a$ el carácter «—» para su posterior impresión.

Línea 5Ø: convierte «resultado» a una cadena alfanumérica para poder incluir el signo «—» si la base es menor que cero.

Línea 1ØØ: Dependiendo de si la base es o no negativa, borra la pantalla.

No se ha tenido en cuenta el que el exponente sea par o impar a la hora de mostrar el resultado; el método para tenerlo en cuenta sería el mismo.

```
   5 REM ***PROGRAMA 2***
   6 LET a$=" ": LET r$=a$: LET
 x=0: LET y=0
   7 CLS
  10 INPUT "Base= ";x
  15 INPUT "Exponente= ";y
  20 LET neg=NOT (SGN x+1)
  30 IF neg THEN PRINT PAPER 5;
INK 1; FLASH 1;"Base negativa,no
rmalmente se ob-tendria Invalid
Argument": PRINT : LET x=ABS x:
LET a$=STR$ (-neg): LET a$=a$(1)
  40 LET resultado=x↑y
  50 LET r$=(a$(1) AND neg)+STR$
 resultado
  60 PRINT "Resultado de x↑y ";
INVERSE 1;r$
  70 PRINT "Pulsa una tecla": PA
USE 0
 100 GO TO (7 AND neg)+(10 AND N
OT neg)
```

LAS MIL CARAS

Matías VAQUERO VICENTE

Spectrum 48 K

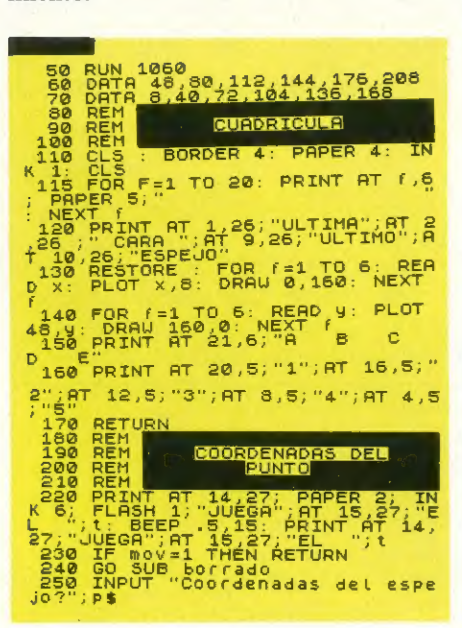
Se trata de un juego de tablero en el que tendremos que demostrar nuestra rapidez de reflejos hasta completar la figura que, paulatinamente, se va dibujando en cada casilla, una vez reflejada en un «espejo».


PEJO

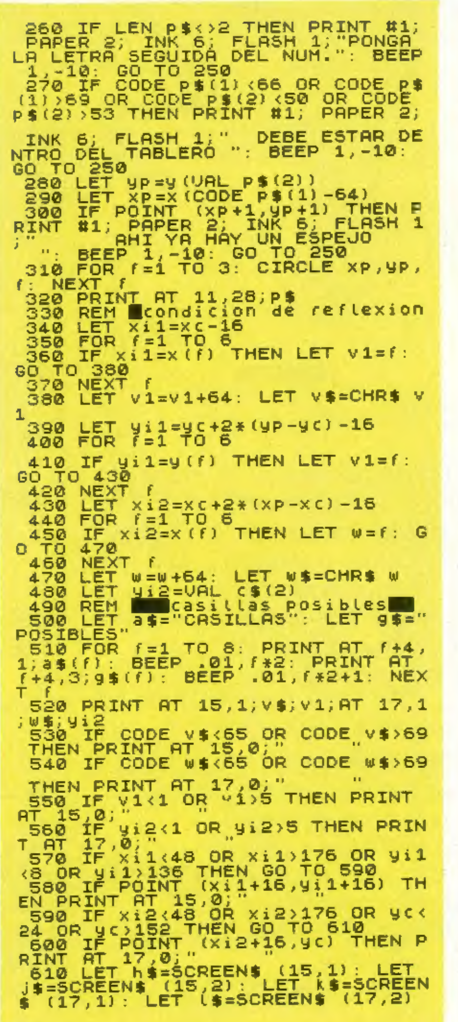
Contando con la participación de dos jugadores, se irá colocando alternativamente una cara en el encasillado que debe cumplir la condición de ser uno de los dos reflejos posibles de la anterior respecto a un punto definido en las intersecciones del interior del tablero.

A medida que el juego avanza, la cara va adquiriendo un rasgo más, hasta que, en un momento determinado, uno de los dos jugadores pierde por no poder colocar un «espejo» o una cara más.

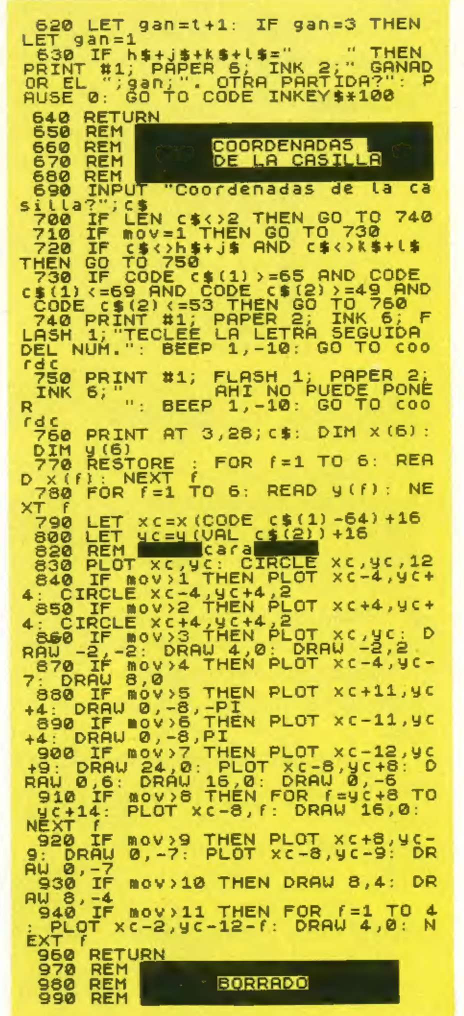
Una peculiaridad de este juego, es la jugada «demostración» automática que aparece en pantalla al cargar la cinta. Esta es bastante similar al juego en sí y nos dará una idea clara de su funcionamiento.



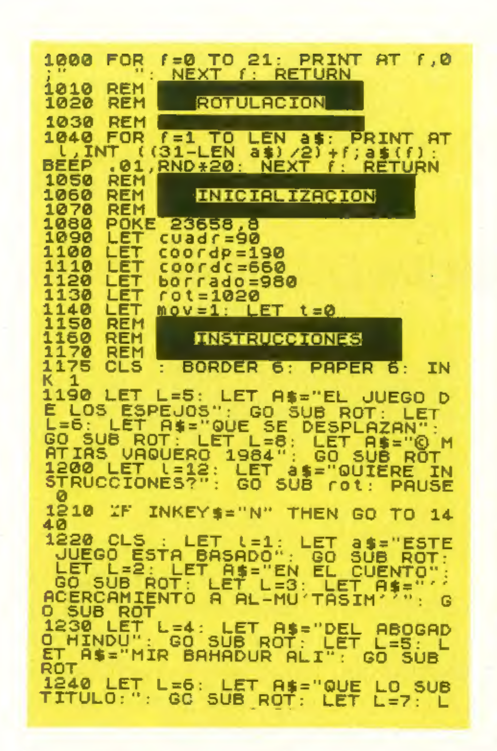
```
  50 RUN 1060
  60 DATA 48,80,112,144,176,208
  70 DATA 8,40,72,104,136,168
  80 REM
  90 REM        CUADRICULA
 100 REM
 110 CLS : BORDER 4: PAPER 4: IN
K 1: CLS
 115 FOR F=1 TO 20: PRINT AT f,6
; PAPER 5;"                    "
: NEXT f
 120 PRINT AT 1,26;"ULTIMA";AT 2
,26 ;" CARA ";AT 9,26;"ULTIMO";A
T 10,26;"ESPEJO"
 130 RESTORE : FOR f=1 TO 6: REA
D x: PLOT x,8: DRAW 0,160: NEXT
f
 140 FOR f=1 TO 6: READ y: PLOT
48,y: DRAW 160,0: NEXT f
 150 PRINT AT 21,6;"A   B   C
D   E"
 160 PRINT AT 20,5;"1";AT 16,5;"
2";AT 12,5;"3";AT 8,5;"4";AT 4,5
;"5"
 170 RETURN
 180 REM
 190 REM     COORDENADAS DEL
 200 REM          PUNTO
 210 REM
 220 PRINT AT 14,27; PAPER 2; IN
K 6; FLASH 1;"JUEGA";AT 15,27;"E
L  ";t: BEEP .5,15: PRINT AT 14,
27;"JUEGA";AT 15,27;"EL  ";t
 230 IF mov=1 THEN RETURN
 240 GO SUB borrado
 250 INPUT "Coordenadas del espe
jo?";p$
```



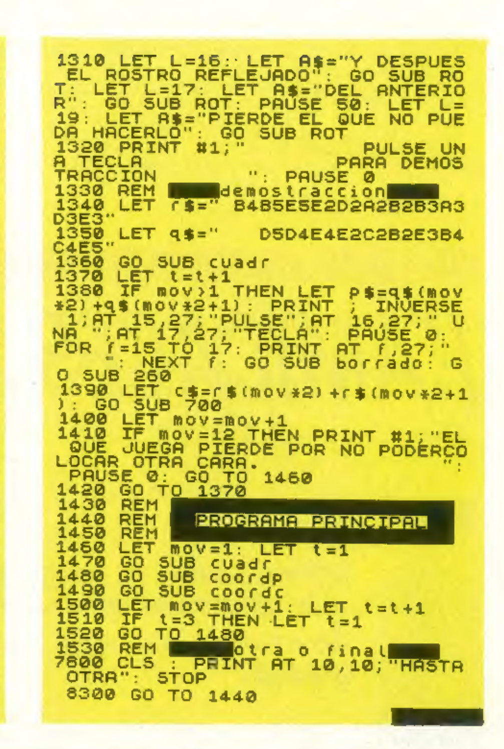
```
 260 IF LEN p$<>2 THEN PRINT #1;
 PAPER 2; INK 6; FLASH 1;"PONGA
LA LETRA SEGUIDA DEL NUM.": BEEP
 1,-10: GO TO 250
 270 IF CODE p$(1)<66 OR CODE p$
(1)>69 OR CODE p$(2)<50 OR CODE
p$(2)>53 THEN PRINT #1; PAPER 2;
 INK 6; FLASH 1;"  DEBE ESTAR DE
NTRO DEL TABLERO ": BEEP 1,-10:
GO TO 250
 280 LET yp=y(VAL p$(2))
 290 LET xp=x(CODE p$(1)-64)
 300 IF POINT (xp+1,yp+1) THEN P
RINT #1; PAPER 2; INK 6; FLASH 1
;"      AHI YA HAY UN ESPEJO
  ": BEEP 1,-10: GO TO 250
 310 FOR f=1 TO 3: CIRCLE xp,yp,
f: NEXT f
 320 PRINT AT 11,28;p$
 330 REM  condicion de reflexion
 340 LET xi1=xc-16
 350 FOR f=1 TO 6
 360 IF xi1=x(f) THEN LET v1=f:
GO TO 380
 370 NEXT f
 380 LET v1=v1+64: LET v$=CHR$ v
1
 390 LET yi1=yc+2*(yp-yc)-16
 400 FOR f=1 TO 6
 410 IF yi1=y(f) THEN LET v1=f:
GO TO 430
 420 NEXT f
 430 LET xi2=xc+2*(xp-xc)-16
 440 FOR f=1 TO 6
 450 IF xi2=x(f) THEN LET w=f: G
O TO 470
 460 NEXT f
 470 LET w=w+64: LET w$=CHR$ w
 480 LET yi2=VAL c$(2)
 490 REM    casillas posibles
 500 LET a$="CASILLAS": LET g$="
POSIBLES"
 510 FOR f=1 TO 8: PRINT AT f+4,
1;a$(f): BEEP .01,f*2: PRINT AT
f+4,3;g$(f): BEEP .01,f*2+1: NEX
T f
 520 PRINT AT 15,1;v$;v1;AT 17,1
;w$;yi2
 530 IF CODE v$<65 OR CODE v$>69
 THEN PRINT AT 15,0;"   "
 540 IF CODE w$<65 OR CODE w$>69
 THEN PRINT AT 17,0;"   "
 550 IF v1<1 OR v1>5 THEN PRINT
AT 15,0;"   "
 560 IF yi2<1 OR yi2>5 THEN PRIN
T AT 17,0;"   "
 570 IF xi1<48 OR xi1>176 OR yi1
<8 OR yi1>136 THEN GO TO 590
 580 IF POINT (xi1+16,yi1+16) TH
EN PRINT AT 15,0;"   "
 590 IF xi2<48 OR xi2>176 OR yc<
24 OR yc>152 THEN GO TO 610
 600 IF POINT (xi2+16,yc) THEN P
RINT AT 17,0;"   "
 610 LET h$=SCREEN$ (15,1): LET
j$=SCREEN$ (15,2): LET k$=SCREEN
$ (17,1): LET l$=SCREEN$ (17,2)
```



```
 620 LET gan=t+1: IF gan=3 THEN
LET gan=1
 630 IF h$+j$+k$+l$="    " THEN
PRINT #1; PAPER 6; INK 2;" GANAD
OR EL ";gan;". OTRA PARTIDA?": P
AUSE 0: GO TO CODE INKEY$*100
 640 RETURN
 650 REM
 660 REM      COORDENADAS
 670 REM      DE LA CASILLA
 680 REM
 690 INPUT "Coordenadas de la ca
silla?";c$
 700 IF LEN c$<>2 THEN GO TO 740
 710 IF mov=1 THEN GO TO 730
 720 IF c$<>h$+j$ AND c$<>k$+l$
THEN GO TO 750
 730 IF CODE c$(1)>=65 AND CODE
c$(1)<=69 AND CODE c$(2)>=49 AND
 CODE c$(2)<=53 THEN GO TO 760
 740 PRINT #1; PAPER 2; INK 6; F
LASH 1;"TECLEE LA LETRA SEGUIDA
DEL NUM.": BEEP 1,-10: GO TO coo
rdc
 750 PRINT #1; FLASH 1; PAPER 2;
 INK 6;"         AHI NO PUEDE PONE
R       ": BEEP 1,-10: GO TO coo
rdc
 760 PRINT AT 3,28;c$: DIM x(6):
 DIM y(6)
 770 RESTORE : FOR f=1 TO 6: REA
D x(f): NEXT f
 780 FOR f=1 TO 6: READ y(f): NE
XT f
 790 LET xc=x(CODE c$(1)-64)+16
 800 LET yc=y(VAL c$(2))+16
 820 REM     cara
 830 PLOT xc,yc: CIRCLE xc,yc,12
 840 IF mov>1 THEN PLOT xc-4,yc+
4: CIRCLE xc-4,yc+4,2
 850 IF mov>2 THEN PLOT xc+4,yc+
4: CIRCLE xc+4,yc+4,2
 860 IF mov>3 THEN PLOT xc,yc: D
RAW -2,-2: DRAW 4,0: DRAW -2,2
 870 IF mov>4 THEN PLOT xc-4,yc-
7: DRAW 8,0
 880 IF mov>5 THEN PLOT xc+11,yc
+4: DRAW 0,-8,-PI
 890 IF mov>6 THEN PLOT xc-11,yc
+4: DRAW 0,-8,PI
 900 IF mov>7 THEN PLOT xc-12,yc
+9: DRAW 24,0: PLOT xc-8,yc+8: D
RAW 0,6: DRAW 16,0: DRAW 0,-6
 910 IF mov>8 THEN FOR f=yc+8 TO
 yc+14: PLOT xc-8,f: DRAW 16,0:
NEXT f
 920 IF mov>9 THEN PLOT xc+8,yc-
9: DRAW 0,-7: PLOT xc-8,yc-9: DR
AW 0,-7
 930 IF mov>10 THEN DRAW 8,4: DR
AW 8,-4
 940 IF mov>11 THEN FOR f=1 TO 4
: PLOT xc-2,yc-12-f: DRAW 4,0: N
EXT f
 960 RETURN
 970 REM
 980 REM        BORRADO
 990 REM
```

```
1000 FOR f=0 TO 21: PRINT AT f,0
;"        ": NEXT f: RETURN
1010 REM
1020 REM     ROTULACION
1030 REM
1040 FOR f=1 TO LEN a$: PRINT AT
 l,INT ((31-LEN a$)/2)+f;a$(f):
BEEP .01,RND*20: NEXT f: RETURN
1050 REM
1060 REM     INICIALIZACION
1070 REM
1080 POKE 23658,8
1090 LET cuadr=90
1100 LET coordp=190
1110 LET coordc=660
1120 LET borrado=980
1130 LET rot=1020
1140 LET mov=1: LET t=0
1150 REM
1160 REM     INSTRUCCIONES
1170 REM
1175 CLS : BORDER 6: PAPER 6: IN
K 1
1190 LET L=5: LET A$="EL JUEGO D
E LOS ESPEJOS": GO SUB ROT: LET
L=6: LET A$="QUE SE DESPLAZAN":
GO SUB ROT: LET L=8: LET A$="© M
ATIAS VAQUERO 1984": GO SUB ROT
1200 LET l=12: LET a$="QUIERE IN
STRUCCIONES?": GO SUB rot: PAUSE
 0
1210 IF INKEY$="N" THEN GO TO 14
40
1220 CLS : LET l=1: LET a$="ESTE
 JUEGO ESTA BASADO": GO SUB ROT:
 LET L=2: LET A$="EN EL CUENTO":
 GO SUB ROT: LET L=3: LET A$="''
ACERCAMIENTO A AL-MU'TASIM''": G
O SUB ROT
1230 LET L=4: LET A$="DEL ABOGAD
O HINDU": GO SUB ROT: LET L=5: L
ET A$="MIR BAHADUR ALI": GO SUB
ROT
1240 LET L=6: LET A$="QUE LO SUB
TITULO:": GO SUB ROT: LET L=7: L
ET A$="''UN JUEGO CON ESPEJOS":
GO SUB ROT: LET L=8: LET A$="QUE
 SE DESPLAZAN''": GO SUB ROT
1250 PAUSE 100: LET L=10: LET A$
="APARECE COMENTADO EN LA REVIST
A": GO SUB ROT: LET L=11: LET A$
="'CACUMEN' NUMERO 20": GO SUB R
OT
1260 PAUSE 100: LET L=13: LET A$
="EL CUENTO (IGUAL QUE EL JUEGO)
": GO SUB ROT: LET L=14: LET A$=
"TRATA DE LA BUSQUEDA DEL ALMA":
 GO SUB ROT: LET L=15: LET A$="D
E ALMOTASIM A TRAVES DE LOS": GO
 SUB ROT: LET L=16: LET A$="REFL
EJOS QUE DEJA EN OTROS": GO SUB
ROT: LET L=17: LET A$="PERSONAJE
S": GO SUB ROT
1270 PRINT #1;"   PULSE UNA TECL
A PARA SEGUIR   ": PAUSE 0: CLS :
 LET L=1: LET A$="EN CADA PASO S
E FORMA UNA CARA": GO SUB ROT: L
ET L=2: LET A$="(UN POCO OCCIDEN
TALIZADA)": GO SUB ROT: LET L=3:
 LET A$="QUE ES EL REFLEJO DE OT
RA": GO SUB ROT: LET L=4: LET A$
="ANTERIOR": GO SUB ROT
1280 LET L=5: LET A$="HASTA QUE
LLEGA EL MOMENTO": GO SUB ROT: L
ET L=6: LET A$="EN QUE UN ROSTRO
 NO SERA REFLEJO": GO SUB ROT: L
ET L=7: LET A$="DE NINGUN OTRO:"
: GO SUB ROT: LET L=8: LET A$="E
S ALMOTASIM": GO SUB ROT
1290 PAUSE 50: LET L=10: LET A$=
"LA PARTIDA Y LA NOVELA": GO SUB
 ROT: LET L=11: LET A$="HAN TERM
INADO": GO SUB ROT
1300 PAUSE 50: LET L=13: LET A$=
"PARA JUGAR SE DEBERA SITUAR": G
O SUB ROT: LET L=14: LET A$="PRI
MERO UN ESPEJO EN LAS": GO SUB R
OT: LET L=15: LET A$="INTERSECCI
ONES DE LAS CASILLAS": GO SUB RO
T
```



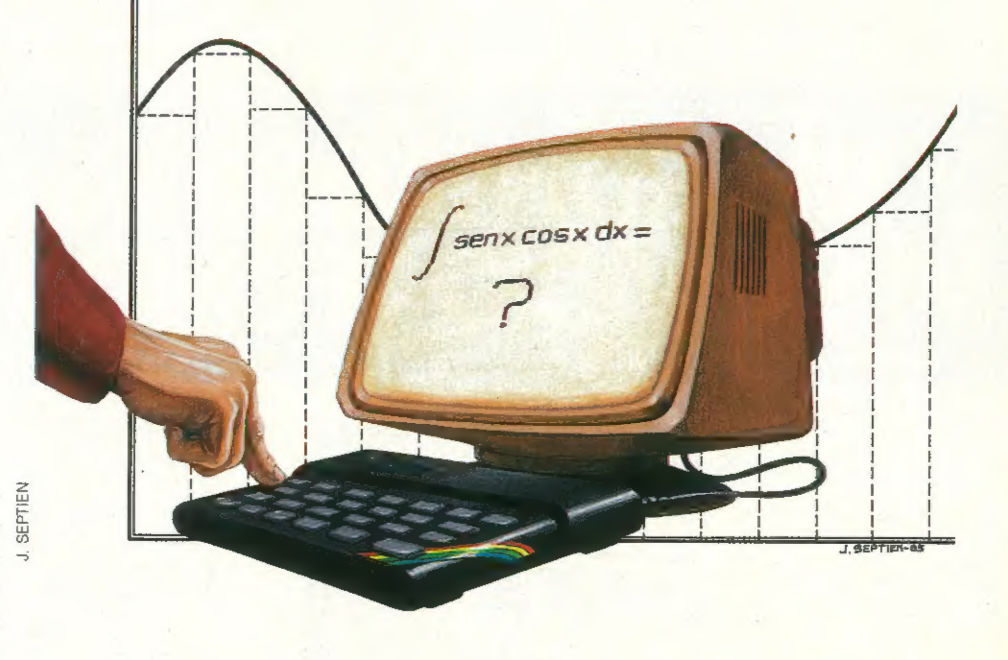
```
1310 LET L=16: LET A$="Y DESPUES
 EL ROSTRO REFLEJADO": GO SUB RO
T: LET L=17: LET A$="DEL ANTERIO
R": GO SUB ROT: PAUSE 50: LET L=
19: LET A$="PIERDE EL QUE NO PUE
DA HACERLO": GO SUB ROT
1320 PRINT #1;"            PULSE UN
A TECLA            PARA DEMOS
TRACCION          ": PAUSE 0
1330 REM     demostraccion
1340 LET r$=" B4B5E5E2D2A2B2B3A3
D3E3"
1350 LET q$="   D5D4E4E2C2B2E3B4
C4E5"
1360 GO SUB cuadr
1370 LET t=t+1
1380 IF mov>1 THEN LET p$=q$(mov
*2)+q$(mov*2+1): PRINT ; INVERSE
 1;AT 15,27;"PULSE";AT 16,27;" U
NA ";AT 17,27;"TECLA": PAUSE 0:
FOR f=15 TO 17: PRINT AT f,27;"
    ": NEXT f: GO SUB borrado: G
O SUB 260
1390 LET c$=r$(mov*2)+r$(mov*2+1
): GO SUB 700
1400 LET mov=mov+1
1410 IF mov=12 THEN PRINT #1;"EL
 QUE JUEGA PIERDE POR NO PODERCO
LOCAR OTRA CARA.              ":
 PAUSE 0: GO TO 1460
1420 GO TO 1370
1430 REM
1440 REM    PROGRAMA PRINCIPAL
1450 REM
1460 LET mov=1: LET t=1
1470 GO SUB cuadr
1480 GO SUB coordp
1490 GO SUB coordc
1500 LET mov=mov+1: LET t=t+1
1510 IF t=3 THEN LET t=1
1520 GO TO 1480
1530 REM     otra o final
7800 CLS : PRINT AT 10,10;"HASTA
 OTRA": STOP
8300 GO TO 1440
```

INTEGRALES

Belén SAENZ DE SANTAMARIA

Spectrum 16 K

El método de las parábolas es, por delante del método de los rectángulos y del de los trapecios, el método de integración aproximada más exacto.



J. SEPTIEN

Geométricamente, una integral definida representa el área comprendida entre la función que queremos integrar, el eje y las rectas. El método de integración aproximada de las parábolas (o método de Simpson) divide el intervalo de integración en n partes iguales (siendo n un número par) y sustituye los trozos de la curva que queremos integrar por arcos de parábola.

Aclarado esto, y una vez tecleado el programa, hemos de proceder a la introdución de los datos de la integral que se quiere realizar, datos tales como extremos de integración, función que se quiere integrar, número de divisiones que se quieren hacer en el intervalo..., que son solicitados por el ordenador mediante sentencias Input.

```
   1 REM © BELEN SAENZ DE SANTAM
ARIA
   2 BORDER 4: INK 4: CLS
   3 PRINT AT 5,6;"INTEGRALES DE
FINIDAS"
   4 PRINT AT 7,5;"METODO DE LAS
 PARABOLAS";AT 8,14;"PARA";AT 9,
5;"INTEGRACION APROXIMADA."
   5 PAUSE 50: BORDER 5: INK 1:
CLS
  10 PRINT AT 3,3;"INTERVALO DE
INTEGRACION"
  20 INPUT "a",a: INPUT "b",b
  30 PRINT AT 5,10;"[a,b]=[";a;"
,";b;"]"
  40 PRINT AT 7,3;"FUNCION A INT
EGRAR"
  50 INPUT "INTEGRAL ";I$
  60 PRINT AT 9,10;I$
  70 PRINT AT 11,3;"NUMERO DE DI
VISIONES"
  80 INPUT "n(par)",n
  90 IF n<=0 OR n-INT n<>0 OR (n
/2)-INT (n/2)<>0 THEN GO TO 80
 100 PRINT AT 13,10;"n=";n
 110 PRINT AT 17,0;"Si quieres e
fectuar alguna","correccion, pul
sa <b>.","Para continuar pulsa",
"cualquier tecla."
 120 PAUSE 0: CLS
 130 IF INKEY$="b" OR INKEY$="B"
 THEN GO TO 10
 140 LET dx=1: LET e=2.71828182
 150 DIM y(n)
 160 LET h=(b-a)/n
 170 FOR k=1 TO n-1
 180 LET x=a+k*h
 190 LET y(k)=VAL I$
 200 NEXT k
 210 LET x=a: LET y0=VAL I$
 220 LET x=b: LET y(n)=VAL I$
 230 LET S=y0+y(n)
 240 FOR k=1 TO n-1 STEP 2
 250 LET S=S+4*y(k)
```

```
260 NEXT k
270 FOR k=2 TO n-2 STEP 2
280 LET S=S+2*y(k)
290 NEXT k
300 LET S=(h/3)*S
310 PRINT AT 5,3;"LA SOLUCON DE
"
320 PRINT AT 6,3;"LA INTEGRAL D
E ";I$
330 PRINT AT 7,3;"ENTRE ";a;" Y
";b;" ES"
340 PRINT AT 10,5;S
350 PRINT AT 21,0;"(Pulse una t
ecla para continuar)"
360 PAUSE 0: CLS
370 PRINT AT 5,3;"ALGUNA INTEGR
AL MAS?"
380 PAUSE 0: CLS : IF INKEY$="S
" OR INKEY$="s" THEN GO TO 10
390 STOP
```

KAMICACES 2.000

Angel Luis MARTINEZ COB

NOTAS GRAFICAS

A B C D E F G H I J K L M N O P
Q R

Spectrum 48 K

Estamos ante un juego de habilidad y destreza en el que tendremos que superar la amenaza de infinidad de naves «Kamicaces» enemigas que nos atacan sin piedad.

Ronda el año 2000 y pilotamos una nave, en un determinado sistema galáctico, atacado constantemente por otras naves enemigas a las que tendremos que disparar antes que ellas nos alcancen. Para ello disponemos de cuatro cañones (cada uno de los cuales se activa con la determinada tecla del cursor) que nos iran consumiendo combustible a medida que los activamos.

Para reponerlo antes de que nos precipitemos al vacio, tendremos que abatirla en una de las veces que atraviesa la pantalla por la parte superior.

Dos datos más: cuando consigamos 2.000 puntos, obtendremos una nave extra, y cada cincuenta naves enemigas que derribemos, pasaremos a una nueva fase con un nivel de dificultad mayor, ya que los kamicaces aparecerán más cerca de nuestra nave.

Para disparar, hay cuatro teclas:

5, hacia la izquierda.
6, hacia abajo.
7, hacia arriba.
8, hacia la derecha.


```
10 RESTORE : BORDER 1: PAPER 1
: INK 7: CLS
30 PRINT AT 4,9;"KAMIKACES 200
0";AT 16,6;"© Martinez Cob. 1984
"
40 FOR i=0 TO 151: READ a: POK
E USR "a"+i,a
42 BEEP .01,a/5
45 PRINT AT 21,2;"▲ ■ |- ■
■ ■ ▲▲ * * .  ": NEXT i: BEEP
1,0
50 DATA 16,16,16,56,56,56,124,
124,0,1,7,255,7,1,2,31
60 DATA 198,255,255,255,255,17
,0,1,0,0,192,254,192,0,128,240
70 DATA 16,16,16,16,16,16,16,1
6,0,0,0,255,0,0,0,0
80 DATA 7,8,16,127,61,15,19,96
,224,16,8,254,188,240,200,6
90 DATA 124,84,68,84,254,254,1
86,146,255,195,223,199,223,255,1
95,24
100 DATA 0,128,192,240,248,252,
254,255,0,1,3,7,7,31,63,255
110 DATA 0,0,0,0,3,15,127,255,3
,7,63,255,255,255,255,255
120 DATA 192,224,248,254,255,25
5,255,255,36,60,255,110,118,255,
60,36
130 DATA 0,36,110,56,28,118,36,
0,0,0,0,24,24,0,0,0
140 DATA 0,0,0,0,0,0,0,0
150 LET r$="BENITO": LET r=100
160 PAPER 0: INK 7: BORDER 0: C
LS : PRINT AT 0,9;"KAMIKACES 200
0"
170 PRINT AT 2,3;"Intenta destr
uir el mayor nu-mero de naves en
emigas.  La navedel fuel te repo
ndra combustiblecuando la derrib
es."
175 PRINT AT 6,3;"A los 2000 pu
ntos nave extra."
180 PRINT AT 8,5;"Dispara con l
as flechas."
190 PRINT INK 3;AT 10,7;"▲";AT
11,6;"■............TU NAVE"
200 PRINT INK 5;AT 13,6;"▲";AT
14,6;"■............COHETE"
210 PRINT INK 6;AT 17,6;"■....
........PLAGAS"
220 PRINT INK 2;AT 20,6;"■....
.....NAVE FUEL"
230 PRINT #1;"  Pulsa una tecla
 para empezar"
232 IF INKEY$="" THEN BEEP .05,
10: PAUSE 50: GO TO 232
240 BORDER 0: PAPER 1: CLS
250 PLOT 33,152: PLOT 184,144
260 PLOT 24,96: PLOT 168,86
270 PLOT 232,88: PLOT 80,78
280 PLOT 29,46: PLOT 147,127
290 PLOT 144,56: PLOT 229,31: P
LOT 76,128
300 PRINT AT 16,23;"▲";AT 17,5
;"▲";AT 17,22;"▲▲";AT 18,13;"
▲▲"
310 INK 4: PRINT AT 18,4;"■ ■
■";AT 18,20;"■  ■";AT 18,31;"
■"
320 PRINT AT 19,0;"■■■■■■■■■■■■
■■■■■■■■■■■■■■■■"
330 PRINT AT 20,0;"■■■■■■■■■■■■
■■■■■■■■■■■■■■■■"
340 PRINT PAPER 7;AT 18,5;"■■";
AT 18,22;"■■■";AT 19,13;"■■■"
350 LET p=0: LET l=1: LET z=0:
LET t=3: LET j=1
360 PRINT INK 0; PAPER 4;AT 21,
0;"Fuel:          00000 Fase:00
▲▲▲"
370 PRINT #1;"Record:";r;TAB 32
-LEN r$;r$
380 LET f=64: FOR i=0 TO 30: NE
XT i
390 BEEP .2,16: BEEP .2,15: BEE
P .2,16: BEEP .21,15: BEEP .22,1
6: BEEP .2,11: BEEP .23,14: BEEP
 .2,12: BEEP 1.2,9: LET m=0
395 INK 0: PAPER 8: PLOT 40,2:
DRAW 64,0: PLOT 40,3: DRAW 64,0
400 FOR i=0 TO 30: NEXT i: LET
k=6
410 FOR i=19 TO 8 STEP -1: INK
3
420 PRINT AT i-2,15;"▲"
430 IF i<19 THEN PRINT AT i-1,1
4;"■"
440 IF i<18 THEN PRINT AT i,14;
"   "
450 BEEP .03,i: NEXT i
460 INK 0: PRINT AT 21,27-LEN S
TR$ l;l;AT 21,32-t;" "
470 PRINT PAPER 0;AT 7,l-1;" ";
AT 7,32-l;" "
480 IF f<=0 THEN GO TO 2100
490 GO SUB 2200
500 IF j THEN GO SUB 1800
510 LET d=1
520 LET q=INT (RND*4)
530 GO TO q*200+600
600 REM Ataque cohete
610 LET s=8: LET x=15
620 FOR y=18 TO 9 STEP -1
630 INK 5: PRINT AT y-1,x;"▲"
640 IF y<18 THEN PRINT AT y,x;"
■"
650 IF y<17 THEN PRINT AT y+1,x
;" "
660 IF INKEY$="6" THEN LET d=0:
 LET f=f-1
670 IF d THEN GO TO 1700
680 PRINT INK 7;AT s,15;"|"
690 IF s=y-1 OR s=y THEN LET p=
p+5: GO TO 1500
700 LET s=s+1: PRINT AT s-1,15;
" "
710 GO TO 1700
800 REM Ataque plaga izquierda
810 LET s=13: LET y=7
820 FOR x=l TO 12
830 PRINT INK 6;AT y,x-1;" ■"
840 IF INKEY$="5" THEN LET d=0:
 LET f=f-1
850 IF d THEN GO TO 1710
860 PRINT INK 7;AT 7,s;"-"
870 IF s=x+1 OR s=x THEN LET p=
p+10: GO TO 1500
880 LET s=s-1: PRINT AT 7,s+1;"
"
890 GO TO 1710
1000 REM Ataque plaga derecha
1010 LET s=17: LET y=7
1020 FOR x=31-l TO 18 STEP -1
1030 PRINT INK 6;AT y,x-1;"■ "
1040 IF INKEY$="8" THEN LET d=0:
 LET f=f-1
1050 IF d THEN GO TO 1710
1060 PRINT INK 7;AT 7,s;"-"
1070 IF s=x+1 OR s=x THEN LET p=
p+10: GO TO 1500
1080 LET s=s+1: PRINT AT 7,s-1;"
"
1090 GO TO 1710
1200 LET v=INT (RND*3)
1202 IF v=0 THEN GO TO 600
1204 IF v=1 THEN GO TO 1400
1210 LET s=5: LET y=1
1220 FOR x=1 TO 16
1230 PRINT INK 2;AT y,x-1;" ■"
1240 IF INKEY$="7" THEN LET d=0:
 LET f=f-1
1250 IF d THEN GO TO 1300
1260 PRINT INK 7;AT s,15;"|"
1270 IF s=1 AND x>13 AND x<16 TH
EN GO TO 1600
1280 IF s=1 THEN PRINT AT 1,15;"
 ": LET d=1: LET s=5: GO TO 1300
1290 LET s=s-1: PRINT AT s+1,15;
" "
1300 NEXT x: PRINT AT y,x-1;"
 ": GO TO 480
1400 REM Nave fuel derecha
1410 LET s=5: LET y=1
1420 FOR x=29 TO 13 STEP -1
1430 PRINT INK 2;AT y,x-1;" ■ "
1440 GO TO 1240
1500 REM Explosiones
1510 FOR e=0 TO 3
1520 INK ABS ((e*4+2)-8)
1530 PRINT AT y-1,x-1;"   ";AT y
,x-1;" ";CHR$ (159+e);" "
1540 PRINT AT y+1,x-1;"   "
1550 BEEP .02,e: NEXT e
1560 LET z=z+1: LET p=p+10
1565 IF INT (z/50)=z/50 THEN GO
TO 1900
1570 GO TO 480
1600 LET f=f+25-l: IF f>64 THEN
LET f=64
1610 INK 0: PLOT 40,2: DRAW f,0:
 PLOT 40,3: DRAW f,0
1620 LET p=p-10: GO TO 1500
1700 NEXT y: GO TO 1720
1710 NEXT x
1720 FOR e=0 TO 3
1730 INK ABS ((e*4+2)-8)
1740 PRINT AT k,15;" "
1750 PRINT AT k+1,12;"   ";CHR$
(159+e);"   "
1760 IF k=6 THEN PRINT AT k+2,15
;" ";AT k+3,15;" "
1770 BEEP .04,e: NEXT e
1780 LET t=t-1: IF NOT t THEN GO
 TO 2500
1790 GO TO 380
1800 IF p<2000 THEN RETURN
1810 LET j=0: LET t=t+1
1820 FOR n=0 TO 10: BEEP .15,20:
 NEXT n
1830 PRINT INK 0;AT 21,33-t;"▲"
1840 RETURN
1900 REM Nueva fase
1910 FOR i=0 TO 5
1920 PRINT INK 7;AT i,15;"|"
1930 BEEP .08,i: NEXT i
1940 FOR i=8 TO 0 STEP -1: INK 3
1950 IF i>1 THEN PRINT AT i-2,15
;"▲"
1960 IF i>0 THEN PRINT AT i-1,14
;"■"
1970 PRINT AT i,14;"   "
1980 BEEP .03,i: NEXT i: LET l=l
+1
1990 FOR i=1 TO f
2000 LET p=p+5: LET f=f-1
2010 GO SUB 2200
2020 IF j THEN GO SUB 1800
2030 BEEP .03,10: NEXT i
2040 BEEP .5,20: GO TO 380
2100 REM Fuel agotado
2110 PRINT AT 21,5;" "
2120 FOR i=5 TO 15: INK 3
2130 PRINT AT i,15;" "
2140 PRINT AT i+1,14;" ▲ ";AT i+
2,14;"■"
2150 BEEP .02,i: NEXT i
2160 LET k=16: GO TO 1720
2200 REM Impresion de puntos
2210 INK 0: PRINT AT 21,19-LEN S
TR$ p;p
2220 INVERSE 1: PLOT 40+f,2: DRA
W 7,0: PLOT 40+f,3: DRAW 7,0: IN
VERSE 0
2230 RETURN
2500 REM Fin de juego
2510 FOR x=0 TO 30: NEXT i
2520 BEEP .3,10: BEEP .3,6: BEEP
 .32,10: BEEP .31,16: BEEP .6,20
2530 INK 7: PRINT FLASH 1;AT 5,5
;"Juego terminado."
2540 PRINT AT 8,1;"Dejaste viuda
s a ";z;" mujeres."
2550 IF p>r THEN GO SUB 2610
2560 PRINT AT 11,1;"Pulsa 'J' pa
ra jugar otra vez.";AT 14,1;"Si
no sabes jugar pulsa 'I'."
2570 POKE 23658,8
2580 IF INKEY$="J" THEN GO TO 24
0
2590 IF INKEY$="I" THEN GO TO 16
0
2600 GO TO 2580
2610 PRINT FLASH 1;AT 11,1;"Escr
ibe tu nombre, campeon!"
2620 POKE 23658,8
2630 IF INKEY$="" THEN BEEP .2,1
0: GO TO 2630
2640 INPUT LINE r$
2650 IF LEN r$>17 THEN LET r$=r$
( TO 17)
2680 LET r=p: RETURN
9999 SAVE "KAMIKACES" LINE 10
```



Chips & Tips

MADRID
Puerto Rico, 21-23
28016 MADRID
Tels. (91) 250 74 04/02

MADRID
Padre Huidobro, s/n.
Ctra. La Coruña, km. 9,2
28023 MADRID
Tel. (91) 207 03 20

VALLADOLID
Juan de Juni, 3
VALLADOLID
Tel. (983) 33 40 00

BILBAO
Alameda Urquijo, 63
48013 BILBAO
Tel. (94) 431 96 67



Sinclair ZX Spectrum*

Nº1 en Ventas

- Color, sonido...
- Alta resolución gráfica
- 2 versiones:

16K RAM 32.000 pts.
48K RAM 41.900 pts.

+ regalo sorpresa

* Garantía INVESTRONICA

ZX Spectrum PLUS*

NUEVO



- 64 K Memoria: 16 K ROM - 48 K RAM
- Color - Sonido
- Teclado profesional (similar al QL)
- 58 teclas, barra espaciadora
- Botón de reset
- Total compatibilidad Software Spectrum
- Total compatibilidad periféricos Spectrum

* Garantía INVESTRONICA

Programas

SPECTRUM

MATCH POINT	1.900
SABRE WULF	2.300
FULL THROTTLE	1.700
TORNADO LOW LEVEL	1.700
OLYMPICON	1.700
STOP THE EXPRESS	1.900
CODE NAME MAT	1.700
BEACH HEAD	1.900
FIGHTER PILOT	1.650
JET SET WILLY	1.650
ANDROID TWO	1.650
ATIC ATAC	1.650
MANIC MINER	1.650
SCUBA DIVE	1.650
ALCHEMIST	1.650
LUNAR JETMAN	1.650
PEDRO	1.650
HUNTER KILLER (Simulador Sub.)	1.650
MASTERCHESS	1.650
LA PULGA	1.900
HORMIGAS	1.900
FRED	1.900
HOMBRE DE NIEVE	1.900
BANDERA A CUADROS	2.500
HORACIO ESQUIADOR	2.000
DEATHCHASE	1.650
1, 2, 3... RESPONDA OTRA VEZ	2.925
BASE DE DATOS	2.200
VU-CALC	2.500
VU-3 D	2.500
COMPILER	2.500
FORTH	2.500
MELODIAN	3.800
FICHERO MUSICAL	2.500
INDIANA JONES	
FIGTHER PILOT	
GHOST BUSTERS	3.700
COLOSSUS CHESS	2.800
ASTRO CHASE	2.300
ZAXXON	2.300
H.E.R.O.	2.800
JET SET WILLY	1.800
KONG	1.700
LA PULGA	1.900
REVENGE MUTANT CAMELS	2.200
CHINESSE JUGLER	1.900
HUSTLER	1.900
PROCESADOR DE TEXTOS	3.300
MAILING ETIQUETAS	3.300
CONTABILIDAD PERSONAL	3.300
BASE DE DATOS	3.300

COMMODORE 64

BEACH HEAD	2.300
WIMBLEDON 64	2.300
MANIC MINER	1.800
SOLO FLIGHT (Simulador vuelo)	3.900
ENCOUNTER	2.300
DECATHLON	2.800
PITFALL	2.800

SPECTRAVIDEO

MUSIC MENTOR	4.900
SECTOR ALPHA	4.900
FRANTIC FREEDY	3.500
SUPER CROSS FIRE	3.500
JUNO LANDER	1.800
FLIPPER SLIPER	3.500
OLD MAC FARMER	2.300
TETRA HORROR	2.300
TELEBUNNY	2.300
TURBOAT	2.300
SASA	2.300
NINJA	2.300
FONT EDITOR	2.300
SPRITE GENERATOR	2.300
SPECTRA DIARY	2.300
SPECTRA HOME ECONOMIST	2.300
SPECTRA CHECKBOOK	2.300
CONTABILIDAD GENERAL I	15.000
CONTROL STOCKS	12.500
PROFILE + ETIQUETAS	12.500
FORTRAN 80	56.000
COBOL 80	Consultar
VENCIMIENTOS	9.500

AMSTRAD

LA PULGA	1.900
FRED	1.900
SPANNER MAN	1.900
BASE DE DATOS	2.300
PASCAL	5.500
DISEÑADOR DE GRAFICOS	4.900
MANIC MINER	1.900
SIMULADOR DE VUELO	2.300
TENIS	1.900
HUNTER KILLER	1.900
HAUNTED EDGES	1.900
ATOM SMASHER	1.900
LABERINTO DEL SULTAN	1.900
ELECTRO FREDDY	1.900
PUNCHY	1.900
MASTERCHESS	1.900
HARRIER ATTACK	1.900
ROLAND EN EL INFIERNO	1.900
AMSWORD (Proc textos)	2.300
AMSCALC (Hoja Cálculo)	3.900
DEVPAC (Ensamblador/Desensamblador)	4.300

Sinclair QL*

NUEVO



- 128 K Memoria ampliables a 640 K
- Procesador 32 bits (M-68008)
- Teclado profesional
- Salida RGB o monitor monocromo y TV.
- 2 Microdrives de 100 K
- Software incluido: Base de datos, hoja de cálculo, procesador de textos, gráficos

* Garantía INVESTRONICA

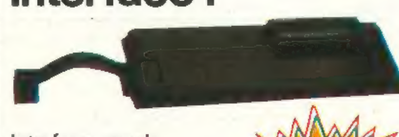
Perifericos Spectrum indescomp*

1. TECLADO MULTIFUNCION
2. CENTRONICS RS/232
3. CONTROLADOR DOMESTICO
4. AMPLIACION DE MEMORIA DE 16 a 48 K EXTERNA
5. AMPLIACION DE MEMORIA DE 16 a 48 K INTERNA
6. AMPLIFICADOR DE SONIDO
7. INTERFACE JOYSTICK TIPO "KEMPSTON"
8. INTERFACE JOYSTICK PROGRAMABLE

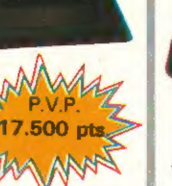


* Garantía INDESCOMP

Interface 1*



Interface para la conexión de hasta 8 unidades Microdrives. Incluye RS-232



P.V.P. 17.500 pts.

* Garantía INVESTRONICA

ZX Microdrive

Unidad de cinta "sinfin" para el almacenamiento rápido de datos. 85 K

P.V.P. 17.500 pts + cartucho blank + manual Learning ¡GRATIS!

* Garantía INVESTRONICA

Wafadrive*

NUEVO



* Garantía MICROBYTE

Sofisticado sistema de almacenamiento rápido de programas y datos.

- 2 Drives
- Catridges de 128 K. c/u
- Cartridges con autoprotección
- Interface RS-232
- Interface Centronics
- Oferta especial de lanzamiento + procesador de texto **Especialmente diseñado para el Spectrum.**

P.V.P. 48.500 pts

AMSTRAD CPC-464*

NUEVO

- 64 K RAM, 32 K ROM
- Alta resolución - gráficos 640×200 pixels
- Superbasic
- Teclado profesional 74 teclas (32 redefinibles)
- Magnetófono incorporado (grabación 1.000 ó 2.000 b.)
- Display de 20, 40, 80 caracteres a voluntad
- Interface paralelo Centronics
- Pantalla monitor incluido





2 versiones
Fósforo verde 12": **89.900 pts.**
Color 14": **126.500 pts.**

* Garantía AMSTRAD

Commodore 64

- 64 K RAM
- Color, sonido
- Alta resolución, manejo de Sprites
- Teclado profesional - teclas de función programable

P.V.P. 54.400 pts. + Manual de programas y 2 programas en cassette ¡GRATIS!

Centronics Commodore

- Permite conectar al CBM-64 cualquier impresora paralelo Centronics (no necesita Software)
- Compatible Software Commodore
- Cable incluido
- Manual de instrucciones en castellano

12.000 pts.

SPECTRAVIDEO SV 328*



* Garantía INDESCOMP

SV-328: 32 K ROM - 80 K RAM
Teclado profesional, color, sonido, alta resolución.
CP/M, Basic Microsoft P.V.P. **67.500 pts.**
+ 1 programa cassette gratis

SV-318: 32 K ROM - 32 K RAM P.V.P. **49.900 pts.**

OFERTA V. PACK SV-328 + Magnetofón + Joystick + 10 cintas P.V.P. **76.000** pts.

SV-728: SISTEMA MSX P.V.P. **64.500** pts.

OFERTA ESPECIAL

- SV-328 ordenador
- SV-605 Expander - Centronics
- Base de Datos
- 1 Caja de Diskettes

P.V.P. 167.400 pts.

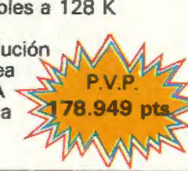
- SV-605 Expander 1 Disco **99.900 pts.**
- SV-605 Expander 2 Discos **148.000 pts.**
- SV-904 Magnetofón **7.900 pts.**
- SV-105 Tableta gráfica **16.800 pts.**
- SV-603 Adaptador juegos Coleco **12.000 pts.**

Apple IIe

TECNICA Y TRADICION



64 K RAM ampliables a 128 K
16 K ROM
Gráficos alta resolución
40 caracteres/línea
Procesador 6502A
La mayor biblioteca de Soft



Apple Macintosh

UNA NUEVA DIMENSION DE ORDENADORES

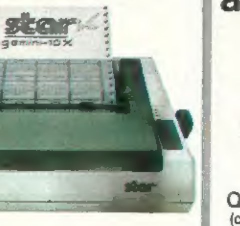


Procesador 32 bits MC 68000 Display 9"
512×342 pixels. 128 K RAM - 64 K ROM
Drive incluido 3½ 400 K
Mouse (Ratón)
Software incluido:
MAC PAINT
MAC WRITE

P.V.P. 583.454 pts

Impresora Star gémini 10X

- 120 c.p.s. bidireccional
- Tracción - fricción
- Papel standard 10"
- Set de caracteres castellano



76.500 pts. + regalo sorpresa

Joysticks alta competición

QUICK-SHOT I
3.400 pts.

QUICK-SHOT II (con disparo automático)
3.900 pts.

ENVIAR A **CHIPS & TIPS**-PUERTO RICO, 21-23. 28016 MADRID,
AUTOBUSES: 7-16-29-51-52 - PEDIDOS POR TEL.: (91) 250 74 04

cantidad	producto	ordenador	ptas.	total

MH

FACILIDADES DE PAGO HASTA 36 MESES SIN ENTRADA
FORMA DE PAGO: □ Contra reembolso □ Incluyo talón nominativo.
D. ....
CALLE ....
NUM. .... PROVINCIA ....

LIBROS
Todos los títulos para todos los ordenadores

PROGRAMAS
Los últimos best-sellers

PERIFERICOS
Impresoras, monitores, diskettes, cintas, etc.

CREDITO
Facilidades de pago hasta 36 meses*
* Solo Madrid

ENVIOS GRATIS A PROVINCIAS

SENSACIONALES OFERTAS PARA USUARIOS DE VIC-20, ZX-81, ORIC, DRAGON, etc.
INFORMESE EN "CHIPS & TIPS"

LOS MAS VENDIDOS

1. Decathlon	Ocean
2. Knight Lore	Ultimate
3. Underwulde	Ultimate
4. Match Point	Psion
5. Sabre Wulf	Ultimate
6. Pyjamarama	Micro Gen
7. Beach Head	U.S. Gold
8. Full Throtle	Micromega
9. Gif From the Goods	Ocean
10. Videolimpic	Dinamic

LISTA ELABORADA CON LA COLABORACION DE:

Micro World
Sinclair Store
Geco Informática
Software Center

Destruye al Robot Zaxxon

ZAXXON

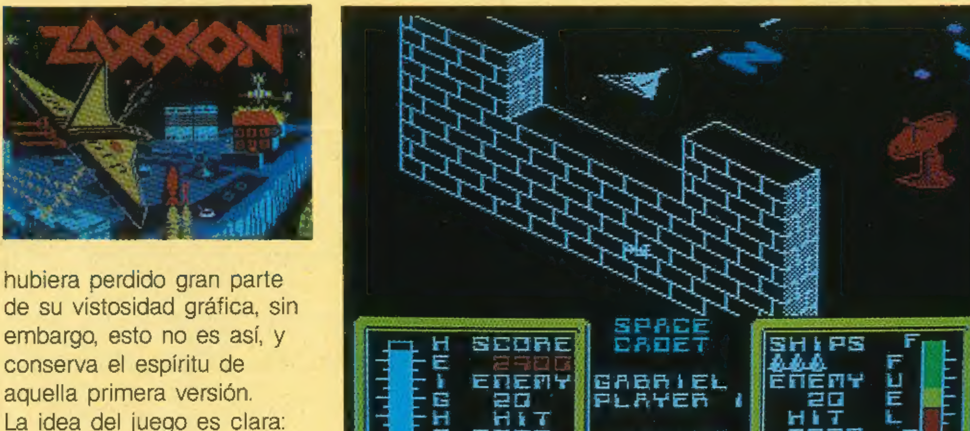
U.S. Gold/ERBE
48 K
Tipo de juego: Arcade
PVP: 2.100



Continuando con la racha bélica que ha hecho tan popular a la casa U.S. Gold, (recordemos si no Beach Head, Blue Max o uno de los últimos Bruce Lee, todavía sólo disponible para el comodore), llega ahora a España el Zaxxon, el popular juego que se hizo tan famoso hace algún tiempo en la versión de Atari, en esta ocasión en su versión para el Spectrum.
Lo lógico era esperar que en el Spectrum, el juego hubiera perdido gran parte de su vistosidad gráfica, sin embargo, esto no es así, y conserva el espíritu de aquella primera versión.
La idea del juego es clara: dirigimos una nave espacial y tenemos que atacar la ciudad asteroide enemiga. En nuestro camino encontraremos emplazamientos con cañones, misiles, aviones enemigos y tanques de combustibles. Estos últimos van a ser muy importantes en el transcurso del juego, ya que si los destruímos podremos reponer combustible.
El movimiento es muy simple y, con Koystick, más aún. Podemos efectuar cuatro movimientos: abajo, arriba, derecha e izquierda, además, por supuesto, del disparo.
Es muy importante que aprendamos a conocer rápidamente el terreno por el que volamos con nuestro aparato, con el fin de aprender a pasar por los huecos de las paredes, por las barreras electrónicas o entre el mortífero fuego de los cohetes.
Hay tres fases dentro del juego: en la primera, tenemos que ir intentando destruir los objetivos que sobrevolamos, mientras atravesamos la ciudad. Si lo conseguimos llegaremos a la segunda fase, en el espacio exterior. Allí nos enfrentaremos a los escuadrones de la aviación enemiga. Si logramos pasar, llegaremos nuevamente a otra ciudad, donde se encuentra el malvado robot Zaxxon, nuestro principal objetivo.







Valoración: Entretenido, con unos gráficos bien construídos y basado en una idea, que si bien no es totalmente original, sí ha sido tratada con mucho mimo.

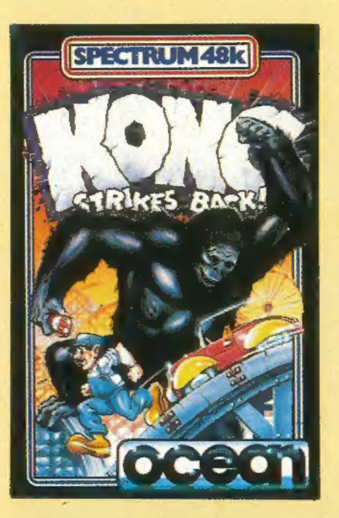
Originalidad	👍👍
Gráficos	👍👍👍👍
Movimiento	👍👍👍👍
Valoración	👍👍👍

Rescate en la montaña rusa

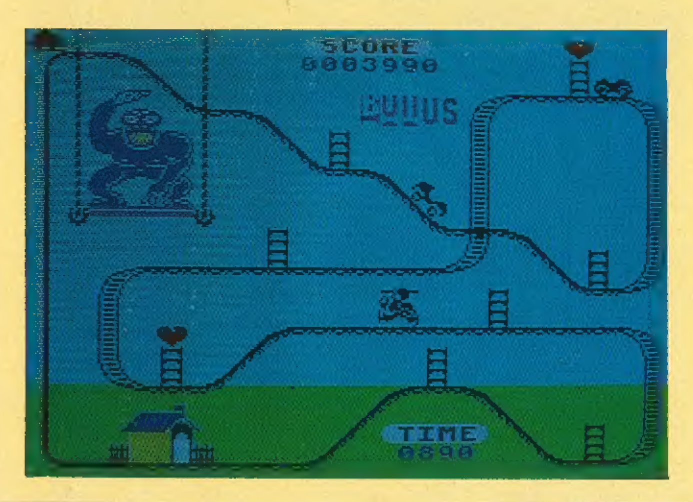
KONG

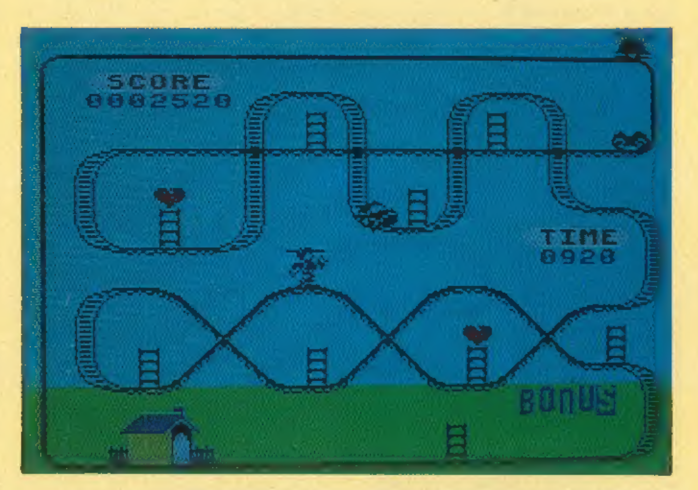
Ocean
48 K
Tipo de juego: Arcade
PVP: Inglés

Nada más comenzar el juego, aparece en la pantalla un parque de atracciones, con una montaña rusa por la que van descendiendo unos peligrosos automóviles que debemos tratar de esquivar, con el fin de evitar que éstos nos destruyan.
En lo alto de la montaña rusa se encuentra una dama, a la cual tenemos que rescatar, pero Kong, el malvado gorila, va a tratar de impedírnoslo. Para conseguir hacernos fracasar en nuestro intento, pone continuamente coches en los raíles de la montaña rusa esperando que estos acaben con nosotros.
Disponemos de cinco vidas y otras tantas bombas que se acabarán si las usamos con demasiada frecuencia y nos veremos en un grave apuro. Afortunadamente, hay dispuestas por todo el recorrido, una serie de escaleras que pueden sernos útiles en caso de encontrarnos en un apuro.
Tenemos que intentar conseguir la máxima puntuación posible, para lo cual es necesario recoger los corazones que hay encima de algunas de las escaleras e intentar llegar adonde se encuentra la dama secuestrada en el menor tiempo posible, sobre todo, si tenemos en cuenta que la relación tiempo-objetivo va a ser la que determine nuestra mayor o menor puntuación.
El movimiento se realiza a través de los raíles; sin embargo, hay que seguir un recorrido preestablecido de modo que no es posible desviarnos en las intersecciones de un carril con otro. Cuando el personaje se dirige hacia abajo, se desliza de la misma forma que si se encontrara en un tobogán.
El movimiento de los coches está bastante logrado, cuando van por los carriles tienen incluso una cierta sensación de suspensión.
Podemos utilizar el iovstick o redefinir las teclas a nuestro gusto, lo que es además muy importante a la hora de facilitar nuestra tarea dentro del juego.











Valoración: Basado en una idea muy original, es un juego que, además de estar muy bien construido, resulta bastante entretenido. Muy recomendable para todo tipo de usuarios.

Originalidad	👍👍👍
Gráficos	👍👍👍
Movimiento	👍👍👍
Valoración	👍👍👍

```
  15 BORDER 4
  20 INPUT "Temporizacion (1-10)
 >>> ";tiempo
  30 IF tiempo<1 OR tiempo>10 TH
EN GO TO 20
  40 LET pausa=tiempo*50
  50 BORDER 1
  60 PAUSE pausa
  70 BORDER 4
  80 PRINT #0;AT 1,0;"Quiere con
tinuar (S/N)"
  90 PAUSE 0
 100 LET a$=INKEY$
 110 IF a$="s" OR a$="S" THEN GO
 TO 20
 120 IF a$="n" OR a$="N" THEN ST
 OP
  130 GO TO 90
```

El valor asignado a la variable «tiempo» se multiplica por 5ɸ para conseguir la temporización deseada, el resultado es asignado a la variable «pausa», utilizada como argumento.

En la línea 9ɸ «PAUSE ɸ» detiene la ejecución del programa hasta que se pulse la tecla «S» o «N».

PROGRAMAS

El programa número «1» simula una máquina de escribir. los controles son:

— SYMBOL SHIFT + Q, borra el carácter anterior.

— SYMBOL SHIFT + W, provoca un retorno de carro.

— SYMBOL SHIFT + E, termina la edición de la página.

Una vez editada la página, puede sacarse una copia por impresora seleccionando la opción «L» o grabarse en cinta con «G». La opción «C» permite editar una nueva página o terminar.

La estructura es la siguiente:

1ɸ : Comentario con el nombre del programa.

2ɸ : Asignación del color rojo para borde, azul para el fondo y blanco para los caracteres.

3ɸ : Inicialización del cursor en la posición ɸ, ɸ.

4ɸ : Visualización del cursor. Se utiliza uno de los gráficos predefinidos.

5ɸ : Pausa hasta que se pulsa una tecla.

6ɸ-1ɸɸ : Verificación de la tecla pulsada. Su código ASCII debe estar comprendido entre el número 32 (espacio) y el 122 (z) o ser uno de los códigos de control del programa:

" < =" (SYMBOL SHIFT + Q)
" < > " (SYMBOL SHIFT + W)
" > =" (SYMBOL SHIFT + E)

11ɸ Visualización de la tecla pulsada.

12ɸ-14ɸ : Incremento de la posición del cursor y comprobación del final de línea y final de página.

15ɸ : Salto a la línea que visualiza el cursor.

1ɸɸɸ : Empieza la rutina que borra el carácter situado a la izquierda del cursor (control = SYMBOL SHIFT + Q).

1ɸɸ5 : Borra el cursor.

1ɸ1ɸ-1ɸ3ɸ : Decremento de la posición del cursor y comprobación del principio de línea y principio de página.

1ɸ4ɸ : Salto a la línea que visualiza el cursor.

11ɸɸ : Comienzo de la rutina que hace saltar el cursor al comienzo de la siguiente línea, simulando el retorno de carro de una máquina de escribir (control = SYMBOL SHIFT + W).

11ɸ5 : Borrado del cursor.

111ɸ : Inicialización de la nueva posición del cursor.

112ɸ : Salto a la línea que comprueba el final de página y línea.

12ɸ5 : Borrado del cursor.

121ɸ : Visualización de las opciones:
L - Listar
G - Grabar
C - Continuar

122ɸ-125ɸ : Comprobación de la opción elegida.

126ɸ-13ɸɸ : Imprimir el contenido de la pantalla. Se utiliza la sentencia «COPY».

131ɸ : Salto a la visualización de las opciones.

135ɸ-138ɸ : Grabar en cinta el contenido de la pantalla. Debe introducirse previamente el nombre que deseamos asignarle. Se utiliza como argumento de «SAVE» la palabra clave «SCREEN$».

139ɸ : Salto a la visualización de opciones.

15ɸɸ-155ɸ : Decisión para editar una nueva página o no.

El programa número «2» permite desplazar, con ayuda de unas teclas utilizadas como cursor, un asterisco (*) a traves de la pantalla, que va dejando un rastro de puntos (.) por donde va pasando.

lor está en función de un «INPUT»; dependiendo de éste la inicialización del puntero y, por tanto, del acceso a los datos.

```
  10 REM ***********
         *         *
         * RESTORE *
         *         *
         ***********

  20 INPUT "Clave (1 a 5) >>> ";
clave
  30 IF clave<1 OR clave>5 THEN
GO TO 20
  40 LET indice=1000+(clave*10)
  50 RESTORE indice
  60 FOR n=1 TO 4
  70 READ codigo
  80 PRINT codigo,
  90 NEXT n
1000 REM
         *********
         *       *
         * DATOS *
         *       *
         *********

1010 DATA 1,2,3,4
1020 DATA 5,6,7,8
1030 DATA 9,10,11,12
1040 DATA 13,14,15,16
1050 DATA 17,18,19,20
```

5ɸɸ DATA	
	3ɸ
	127
	2ɸɸ
51ɸ DATA	
	6ɸ
	32
INDICE → 52ɸ DATA	
	114
	123
	15
	62
53ɸ DATA	
	94

DATOS

Mecanismo "Restore 52ɸ".

Errores

Cuando se manejan sentencias del tipo READ/DATA, hay tres tipos de error que suelen producirse frecuentemente.

a) Cuando se ejecuta una sentencia «READ» y el puntero se encuentra al final de la tabla, indicando que no hay más datos. El mensaje que se visualiza es:

E Out of DATA

Ejemplo:

```
  10 REM ***********
         * ERROR 1 *
         ***********
  20 FOR n=1 TO 5
  30 READ nota
  40 PRINT nota
  50 NEXT n
 100 REM
         *********
         * DATOS *
         *********
 110 DATA 10,23,56,78
```

se pretende leer cinco datos, cuando en realidad sólo hay cuatro.

b) Cuando al confeccionar la tabla de datos se incluye un valor numérico en lugar de uno alfanumérico, o viceversa. Al ejecutarse dicho programa y tratar de leer un dato que no está en corcondancia con el de la variable que acompaña a «READ», se produce el error:

C Nonsense in BASIC

Ejemplo:

```
  10 REM ***********
         * ERROR 2 *
         ***********
```

```
  15 PRINT "NOTA","  ALUMNO"
  16 PRINT "_____","____________"''
  20 FOR n=1 TO 8
  30 READ nota,j$
  40 PRINT nota,j$
  50 NEXT n
 100 REM
         *********
         *       *
         * DATOS *
         *       *
         *********
 110 DATA 5,"ANTONIO",8,"GERARDO
"
 120 DATA 2,"SEBASTIAN",10,"LUIS
"
 130 DATA 7,"CARLOS",9,"PABLO"
 140 DATA 3,4,6,"RAFAEL"
```

En la línea 14ɸ, el segundo dato debería ser alfanumérico.

```
10 REM ***************
       *             *
       * CURSO/BASIC *
       *             *
       ***************
       *             *
       * GEOGRAFIA 1 *
       *             *
       ***************
 20 BORDER 4: PAPER 4: INK 1: C
LS
 30 GO SUB 1000
 40 REM
       *******************
       *                 *
       * BUCLE PRINCIPAL *
       *                 *
       *******************
 50 PRINT AT 1,3;"AUTONOMIAS DE
 LA PENINSULA"
 60 PRINT AT 2,3;"______________
______________"
 70 FOR c=5 TO 19
 80 READ a$
 90 LET opcion=c-4
100 PRINT AT c,3;opcion;
110 IF opcion<10 THEN PRINT " "
;
120 PRINT " - ";a$
130 NEXT c
140 INPUT AT 0,1;"Que Autonomia
desea visualizar  >>> ";opcion
150 IF opcion<1 OR opcion>15 TH
EN GO TO 140
160 CLS
170 LET direccion=2000+(opcion*
100)
180 RESTORE direccion
190 READ a$
200 PRINT AT 1,3;a$
210 PRINT "_____________________
____________"
220 READ numero
230 FOR n=1 TO numero
240 PRINT '" - ";
250 READ a$,b$
260 PRINT a$,"(";b$;")"
270 NEXT n
280 INPUT "Pulse ""ENTER"" para
retornar "; LINE a$
290 RESTORE 1100
300 CLS
310 GO TO 40
1000 REM
       *******************
       *                 *
       * DEFINICION "Ñ"  *
       *                 *
       *******************
1010 RESTORE 1060
1020 FOR d=USR "n" TO USR "n"+7
1030 READ dato
1040 POKE d,dato
1050 NEXT d
1060 DATA 24,66,98,82,74,70,66,0
1070 RETURN
1100 REM
       ********
       *      *
       * MENU *
       *      *
       ********
1110 DATA "ANDALUCIA","ARAGON","
ASTURIAS","CANTABRIA","CASTILLA-
LA MANCHA"
1120 DATA "CASTILLA-LEON","CATAL
UÑA","COMUNIDAD VALENCIANA","EXT
REMADURA","GALICIA"
1130 DATA "MADRID","MURCIA","NAV
ARRA","PAIS VASCO","LA RIOJA"
2000 REM
       ***************
       *             *
       * PROVINCIAS  *
       *             *
       ***************
2100 DATA "ANDALUCIA",8,"ALMERIA
","ALMERIA","CADIZ","CADIZ","COR
DOBA","CORDOBA","GRANADA","GRANA
DA","HUELVA","HUELVA","JAEN","JA
EN","MALAGA","MALAGA","SEVILLA",
"SEVILLA"
2200 DATA "ARAGON",3,"HUESCA","H
UESCA","TERUEL","TERUEL","ZARAGO
ZA","ZARAGOZA"
2300 DATA "ASTURIAS",1,"ASTURIAS
","OVIEDO"
2400 DATA "CANTABRIA",1,"CANTABR
IA","SANTANDER"
2500 DATA "CASTILLA-LA MANCHA",5
,"ALBACETE","ALBACETE","CIUDAD R
EAL","CIUDAD REAL","CUENCA","CUE
NCA","GUADALAJARA","GUADALAJARA"
,"TOLEDO","TOLEDO"
2600 DATA "CASTILLA-LEON",9,"AVI
LA","AVILA","BURGOS","BURGOS","L
EON","LEON","PALENCIA","PALENCIA
","SALAMANCA","SALAMANCA","SEGOV
IA","SEGOVIA","SORIA","SORIA","V
ALLADOLID","VALLADOLID","ZAMORA"
,"ZAMORA"
2700 DATA "CATALUÑA",4,"BARCELON
A","BARCELONA","GERONA","GERONA"
,"LERIDA","LERIDA","TARRAGONA","
TARRAGONA"
2800 DATA "COMUNIDAD VALENCIANA"
,3,"ALICANTE","ALICANTE","CASTEL
LON","C. DE LA PLANA","VALENCIA"
,"VALENCIA"
2900 DATA "EXTREMADURA",2,"BADAJ
OZ","BADAJOZ","CACERES","CACERES
"
3000 DATA "GALICIA",4,"LA CORUÑA
","LA CORUÑA","LUGO","LUGO","ORE
NSE","ORENSE","PONTEVEDRA","PONT
EVEDRA"
3100 DATA "MADRID",1,"MADRID","M
ADRID"
3200 DATA "MURCIA",1,"MURCIA","M
URCIA"
3300 DATA "NAVARRA",1,"NAVARRA",
"PAMPLONA"
3400 DATA "PAIS VASCO",3,"ALAVA"
,"VITORIA","GUIPUZCOA","SAN SEBA
STIAN","VIZCAYA","BILBAO"
3500 DATA "LA RIOJA",1,"LA RIOJA
","LOGROÑO"
```

```
2270 DATA "SELLA","VALLE SAJAMBR
E","LEON",1,"PONGA"
2300 DATA 7,"MEDITERRANEA"
2310 DATA "EBRO","FONTIBRE","SAN
TANDER",7,"ARAGON","EGA","GALLEG
O","GUADALOPE","HUERVA","JALON"
"SEGRE"
2320 DATA "GUADALHORCE","SIERRA
DE GIBALTO","MALAGA-GRANADA",0
2330 DATA "JUCAR","CERRO SAN FEL
IPE","CUENCA",1,"CABRIEL"
2340 DATA "LLOBREGAT","FONT DEL
LLOBREGAT","BARCELONA",2,"CARDON
ER","NOYA"
2350 DATA "SEGURA","SIERRA DEL S
EGURA","JAEN",3,"ARGOS","BENAMOR
","MUNDO"
2360 DATA "TER","MONT COSTA BOND
","GERONA",2,"FRESSER","OÑAR"
2370 DATA "TURIA","MUELA SAN JUA
N","TERUEL",0
```

c) En las tablas de datos con valores de cadena, puede suceder que se nos olvide colocar las correspondientes comillas. El intérprete BASIC, al ejecutarse el programa, tomará dicho dato como variable numérica; si no existe ninguna variable

con dicho nombre, aparecerá el conocido mensaje:

2 Variable not found

si por el contrario existiera, nos encontraríamos en el caso expuesto en el apartado «b».

Ejemplo:

```
10 REM ************
       *          *
       * ERROR 3  *
       *          *
       ************
20 FOR n=1 TO 8
30 READ s$
40 PRINT s$
50 NEXT n
100 REM
```

```
       *********
       *       *
       * DATOS *
       *       *
       *********
110 DATA "AUTOMOVIL","BICICLETA
","AUTOBUS","HELICOPTERO"
120 DATA "MOTO",BARCO,"AVION","
SUBMARINO"
```

En la línea 12∅, el dato BARCO debería ir entre comillas.

PROGRAMA 2

```
 10 REM ***************
        *             *
        * CURSO/BASIC *
        *             *
        ***************
        *             *
        * GEOGRAFIA 2 *
        *             *
        ***************
 20 BORDER 1: PAPER 4: INK 0: C
LS
 30 GO SUB 1000
 40 REM
        *******************
        *                 *
        * BUCLE PRINCIPAL *
        *                 *
        *******************
 50 PRINT AT 1,9;"RIOS ESPAÑOLE
S"
 60 PRINT "______________________
__________"
 65 PRINT AT 5,8;"________________
"
 70 PRINT AT 7,11;"VERTIENTES"
 75 PRINT AT 8,8;"________________
"
 80 PRINT AT 12,8;"1 - ATLANTIC
A"
 90 PRINT AT 15,8;"2 - CANTABRI
CA"
100 PRINT AT 18,8;"3 - MEDITERR
ANEA"
110 INPUT "Que vertiente desea
>>> ";vertiente
120 IF vertiente<1 OR vertiente
>3 THEN GO SUB 1100: GO TO 110
130 LET rios=2000+(vertiente*10
0)
140 RESTORE rios
150 READ numero,a$
160 GO SUB 1200
170 INPUT "Que rio desea >>> ";
rio
180 IF rio<1 OR rio>numero THEN
 GO SUB 1100: GO TO 170
190 LET direccion=n-((numero-ri
o+1)*10)
200 RESTORE direccion
210 CLS
220 GO SUB 1300
230 INPUT "Pulse ""ENTER"" para
 retornar "; LINE z$
240 CLS
250 GO TO 50
1000 REM *******************
         *                 *
         * DEFINICION "Ñ"  *
         *                 *
         *******************
1010 RESTORE 1060
1020 FOR d=USR "n" TO USR "n"+7
1030 READ dato
1040 POKE d,dato
1050 NEXT d
1060 DATA 24,66,98,82,74,70,66,0
1070 RETURN
1100 REM *********
         *       *
         * ERROR *
         *       *
         *********
1110 PRINT #0;">>>>>>>>>>>>> ERRO
R <<<<<<<<<<<<<"
1120 FOR j=1 TO 200: NEXT j
1130 RETURN
1199 REM
         ********
         *      *
         * RIOS *
         *      *
         ********
1200 CLS
1210 PRINT AT 1,5;a$
1220 PRINT "_____________________
__________"
1225 LET X=1
1230 FOR n=rios+10 TO rios+(nume
ro*10) STEP 10
1240 PRINT '" ";X;" - ";
1250 RESTORE n
1260 READ a$
1270 PRINT a$
1275 LET X=X+1
1280 NEXT n
1290 RETURN
1299 REM
         *************
         *           *
         * AFLUENTES *
         *           *
         *************
1300 CLS
1310 READ a$,b$,c$
1320 PRINT AT 1,1;a$
1330 PRINT "_____________________
__________"
1340 PRINT AT 4,1;"Nacimiento: "
;b$
1345 PRINT AT 6,1;"Provincia : "
;c$
1350 READ afluentes
1360 IF afluentes=0 THEN RETURN
1364 PRINT AT 9,9;"_____________"
1365 PRINT AT 11,10;"Afluentes"
1366 PRINT AT 12,9;"_____________"
1370 FOR z=1 TO afluentes
1380 READ d$
1390 PRINT " - ";d$,
1400 NEXT z
1410 RETURN
2000 REM
         *********
         *       *
         * DATOS *
         *       *
         *********
2100 DATA 8,"ATLANTICA"
2110 DATA "DUERO","PEÑA DE URBIO
N","LOGROÑO-SORIA",5,"ADAJA","ER
ESMA","ESLA","PISUERGA","TORMES"
2120 DATA "GUADALQUIVIR","SIERRA
 DE CAZORLA","JAEN",4,"GENIL","G
UADAJOZ","GUADALIMAR","GUADIATO"
2130 DATA "GUADIANA","OJOS DEL G
UADIANA","ALBACETE-C. REAL",4,"R
UECAS","ZANCARA","ZIGUELA","ZUJA
R"
2140 DATA "MIÑO","LAGUNA FUENTE
MIÑO","LUGO",3,"ARNOYA","FERREIR
AS","SIL"
2150 DATA "ODIEL","SIERRA DE ARA
CENA","HUELVA",1,"TINTO"
2160 DATA "TAJO","SIERRA ALBARRA
```


LECTURA DEL TECLADO Y TEMPORIZACIONES

INKEY$

Acceso al tedado

IN KEY $

N NEXT — MODO E

OVER

Tipo de sentencia

Función de entrada.

Definición

La función «INKEY$» permite leer un solo carácter introducido por teclado. Se diferencia básicamente de la sentencia «INPUT» en:

a) INKEY$ no produce *eco*, es decir, que no se visualiza el valor de la tecla pulsada.
b) El dato tiene que estar preparado al ejecutarse la función «INKEY$», es decir, que no espera, como en el caso de «INPUT», a que se teclee el dato: por tanto, si no está preparado se ejecuta la siguiente instrucción y la función retorna una cadena vacía (" ").

«INKEY$» no tiene ningún argumento, por el contrario, forma parte de los argumentos de sentencias del tipo:

— LET.
— PRINT.
— IF... THEN...

Ejemplos:

```
10 REM ************
       *          *
       * INKEY$ 1 *
       *          *
       ************
20 IF INKEY$="" THEN GO TO 20
30 PRINT INKEY$;
40 GO TO 10
```

1) En el siguiente programa se visualiza el valor retornado por la función «INKEY$» cuando se pulsa una tecla, es decir, cuando no es una cadena vacía.
observe que «INKEY$» diferencia entre mayúsculas (modo C), minúsculas (modo L) o aquellos símbolos que se pulsan conjuntamente con SYMBOL SHIFT.

```
10 REM ************
       *          *
       * INKEY$ 2 *
       *          *
       ************
 20 INPUT "Numero >>> ";numero
 30 PRINT numero
 40 PRINT #0;AT 1,0;"Quiere int
roducir mas (s/n)"
 50 IF INKEY$="" THEN GO TO 50
 60 IF INKEY$="s" OR INKEY$="S"
 THEN GO TO 20
 70 IF INKEY$="n" OR INKEY$="N"
 THEN STOP
 80 GO TO 40
```

2) En este caso, «INKEY$» se utiliza para preguntar sobre una tecla determinada. Tiene la ventaja sobre la utilización de «INPUT», en que no es necesario pulsar «ENTER».

```
10 REM ************
       *          *
       * INKEY$ 3 *
       *          *
       ************
```

```
 20 PRINT AT 2,9,"MENU PRINCIPA
L"
 30 PRINT AT 3,9,"_______________
"
 40 PRINT AT 8,10,"1 - EDITAR."
 50 PRINT AT 10,10,"2 - LISTAR.
"
 60 PRINT AT 12,10,"3 - BORRAR.
"
 70 PRINT AT 14,10,"4 - ARCHIVA
R."
 80 PRINT AT 16,10,"5 - INSERTA
R."
 90 PRINT #0,AT 1,0,"Elija la o
pcion (1 - 5)"
100 LET a$=INKEY$
110 IF a$="1" THEN GO TO 1000
120 IF a$="2" THEN GO SUB 2000
130 IF a$="3" THEN GO TO 1500
140 IF a$="4" THEN GO TO 1750
150 IF a$="5" THEN GO SUB 2100
160 GO TO 100
```

3) Cuando el valor retornado por «INKEY$», se utiliza en sentencias posteriores a la pulsación de la tecla, conviene asignar este valor a una variable de cadena.

PAUSE

Acceso al tedado

PI

M PAUSE — MODO K

INVERSE

Tipo de sentencia

Comando de programación.

Definición

Esta sentencia suspende la ejecución de un programa durante un tiempo determinado, es decir, *temporiza* o hace una *pausa*.

El efecto que produce es similar al proporcionado por las sentencias:

Diferencias entre "INKEY$" e "INPUT".

```
1∅ FOR x = 1 TO tiempo
2∅ NEXT x
```

La estructura de este comando es:

SENTENCIA	ARGUMENTO
PAUSE	expresión numérica

Ejemplos:

— PAUSE 1∅.
— PAUSE 95∅.
— PAUSE 3∅ * 5.

Si se introduce un número fraccionario, éste se redondea al valor entero más cercano. El rango de valores está comprendido entre ∅ y 65535, cuando se introduce uno fuera de margen aparece el mensaje.

B Integer out of range

La realización que existe entre el número del argumento y el tiempo de temporización, en segundos es:

$$\text{tiempo} = \frac{n}{5\emptyset}$$

por lo tanto «PAUSE 1∅∅», detiene la ejecución del programa durante «2» segundos. Dividiendo entre 5∅ el valor máximo (65535) obtenemos el mayor tiempo de temporización:

65535/5∅ = 131∅.7 sg.

21 min. 5∅.7 sg.

ATENCION

Si durante la ejecución de una sentencia «PAUSE» se presiona una tecla, la temporización se interrumpe y continúa el programa en la siguiente instrucción.

El valor «∅», asignado a «PAUSE», provoca una temporización indefinida hasta que se pulsa una tecla.

Puede utilizarse una variable del tipo numérico como argumento de esta sentencia.

Ejemplo:

```
10 REM ****************
       *              *
       * PAUSA VARIABLE *
       *              *
       ****************
```

```
CIN","TERUEL",7,"ALAGON","ALBERC
HE","ALMONTE","GUADARRAMA","GUAD
IELA","JARAMA","TIETAR"
2170 DATA "TAMBRE","MONTES DE BO
CELO","LA CORUÑA",2,"SAMO","LENG
UELLE"
2180 DATA "ULLA","MONTE CEBREIRO
","LUGO",3,"DEZA","ISO","SAR"
2200 DATA 7,"CANTABRICA"
2210 DATA "BIDASOA","MAYA","NAVA
RRA",0
```

```
2220 DATA "EO","MONTES DEL CADAB
O","LUGO",0
2230 DATA "NALON","PUERTO TARNA"
,"OVIEDO",4,"CAUDAL","NARCEA","N
ORA","TRUBIA"
2240 DATA "NAVIA","SIERRA RAÑADO
IRO","LUGO",0
2250 DATA "NERVION","PEÑA DE ORD
UÑA","ALAVA",1,"IBAIZABAL"
2260 DATA "ORIA","MONTE AITZGORR
I","GUIPUZCOA",0
```

Programas

Como aplicación de las sentencias «READ/DATA», se presentan, en esta ocasión, dos programas de utilidad didáctica. Ambos están relacionados con un área muy importante dentro de la enseñanza: la Geografía.

El alfabeto español tiene una letra que no está incluida en el juego de caracteres del Spectrum, ésta es la «Ñ»; por tanto ha sido necesario diseñar un nuevo gráfico (GDU) con dicha forma. La técnica empleada para realizarlo será explicada en el capítulo dedicado a gráficos. Para acceder a la «Ñ» es necesario pasar a modo gráfico (G), como recordará, basta con pulsar simultáneamente la tecla «CAPS SHIFT» y «9», una vez en este modo, debe pulsarse la tecla «N».

ATENCION

Hasta que el programa no se ejecute, *no se visualizará* el gráfico correspondiente a la «Ñ», por tanto no se preocupe si ésta no aparece al pasar al modo G.

Para retornar al modo anterior pulse la tecla «9».

El primer programa está dedicado al estudio de las *Autonomías*. Al ejecutar el programa, se presenta un menú con las distintas Autonomías de la Península. Cada una de ellas tiene a su izquierda un número de opción, éste sirve como referencia para seleccionarla. Una vez elegida la opción, aparece en la parte superior de la pantalla el nombre de la Autonomía, y a continuación, el nombre de las provincias que la componen con sus correspondientes capitales, encerradas entre paréntesis.

La estructura del programa es:

- 10 : Comentario con el nombre del programa.
- 20 : Asignación del color verde para fondo y papel y azul para los caracteres.
- 30 : Llamada a la subrutina que define como GDU, la «Ñ».
- 40 : Comienzo del programa principal.
- 50-60 : Rótulo de presentación.
- 70-130 : Bucle para visualizar el número de opción y el nombre de las Autonomías, éstas están contenidas en una tabla de datos.
- 140-150 : Entrada opción y comprobación.
- 160 : Borrado pantalla.
- 170 : *Algoritmo* empleado para el cálculo del número de línea, dónde debe inicializarse el puntero de la tabla de datos.
- 180 : Inicialización del puntero (RESTORE).
- 190-210 : Lectura y visualización del nombre de la Autonomía.
- 220 : Lectura del número de provincias.
- 230-270 : Bucle para la lectura y visualización de las provincias y sus capitales.
- 280 : Entrada de la tecla «ENTER».
- 280 : Inicialización de la tabla.
- 300-310 : Borrado pantalla y regreso al menú principal.
- 1000-1070 : Subrutina para definición de la tecla «Ñ».
- 1100-1130 : Datos del menú principal.
- 2000-3500 : Datos correspondientes a las provincias y sus capitales.

El segundo programa sirve apra estudiar o consultar sobre la hidrografía española. Al ejecutarse presenta en pantalla el menú con las tres vertientes:

- ATLANTICA.
- CANTABRICA.
- MEDITERRANEA.

Al elegirse una de ellas aparece otro menú con los ríos que desembocan. Al seleccionar uno de ellos, se visualiza en pantalla la información correspondiente a:

- Nombre del río.
- Nacimiento.
- Provincia.
- Afluentes más importantes (si los tiene).

Pulsando «ENTER» se retorna al menú principal. Si al elegir las opciones se selecciona una que no existe, aparece en la parte inferior de la pantalla un mensaje de error.

DATOS "Ñ"
AUTONOMIAS DEL "MENU"
AUTONOMIA
N.º DE PROVINCIAS
PROVINCIA
CAPITAL
PROVINCIA
CAPITAL
AUTONOMIA
PROVINCIA
CAPITAL

Estructura de datos del programa "Autonomías".

La estructura del programa es la siguiente:

- 10 : Comentario con el nombre del programa.
- 20 : Asignación del color azul para el borde, verde para el fondo y negro para los carácteres.
- 30 : Llamada a la subrutina que define la «Ñ».
- 40 : Comienzo del programa principal.
- 50-100 : Presentación del menú con las vertientes.
- 110-120 : Entrada de opción y comprobación.
- 130 : Algoritmo que calcula el número de línea donde debe inicializarse la lectura de los ríos.
- 140 : Inicialización del puntero.
- 150 : Leer número de ríos y vertiente.
- 160 : Llamada a la subrutina que lee y visualiza el menú con los nombres de los ríos.
- 170-180 : Entrada de la opción «río» y verificación.
- 190 : Algoritmo para calcular dónde debe inicializarse la lectura de los datos correspondientes al río elegido.
- 200 : Inicialización del puntero.
- 210 : Borrado pantalla.
- 220 : Llamada a la subrutina que lee y visualiza los datos correspondientes al dato elegido.

DATOS "N"
N.º DE RIOS
VERTIENTE
RIO
NACIMIENTO
PROVINCIA
N.º DE AFLUENTES
AFLUENTE
RIO
N.º DE RIOS
VERTIENTE
AFLUENTE
DATOS POR VERTIENTE
DATOS POR RIO

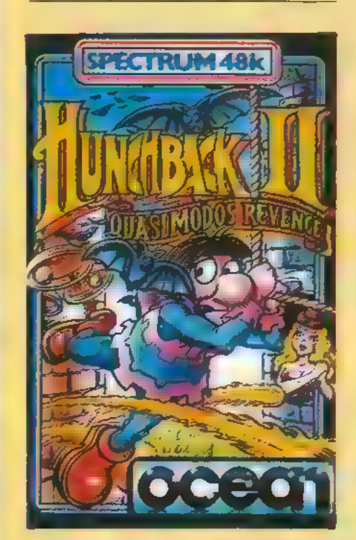
Estructura de datos del programa "Ríos".

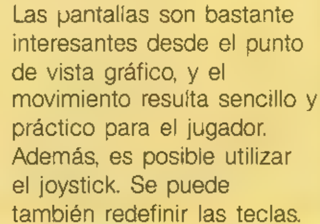
- 230 : Entrada de la tecla «ENTER».
- 240-250 : Borrado pantalla y vuelta al menú principal (vertientes).
- 1000-1070 : Subrutina para definir la letra «Ñ».
- 1100-1130 : Subrutina de presentación del mensaje «ERROR».
- 1200-1290 : Subrutina para leer y visualizar los ríos.
- 1300-1410 : Subrutina para leer y visualizar los datos, (nacimientos, provincia, ...)
- 2000-2370 : Tabla de datos. ■

La venganza de Quasimodo

Hunchback II

Ocean
48 K
Tipo de juego: Arcade
PVP: Inglés.



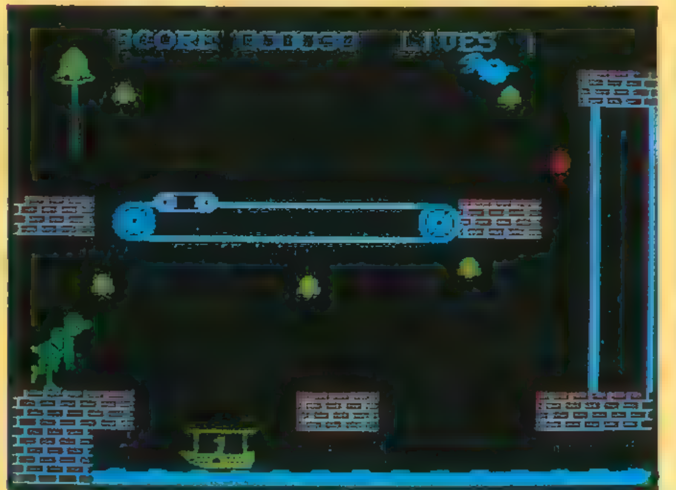


Se trata de la segunda parte de este conocido juego, en el que nuestro amigo Quasimodo vuelve a intentar rescatar a su amada Esmeralda de la peligrosa torre donde se encuentra prisionera.

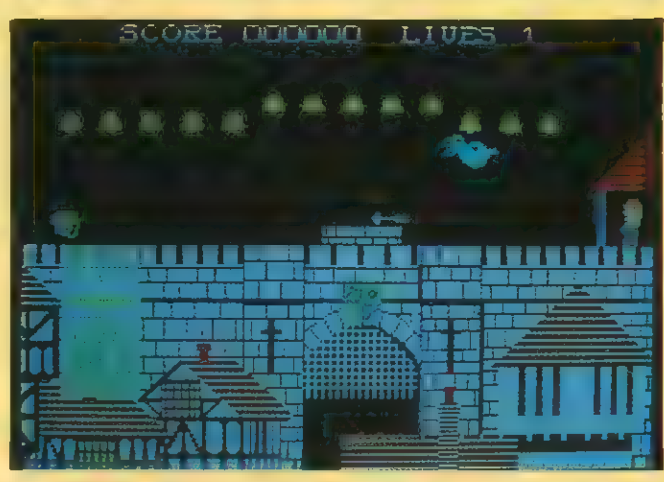
Al igual que en la primera parte, las trampas son muchas y la dificultad bastante alta en todas las fases del mismo; pero esto es quizás un aliciente más para el jugador apasionado, que vera cómo poco a poco van saltando las barreras que se interponen en su camino, barreras que por cierto, en esta ocasión, son más complicadas aún que en la primera versión.

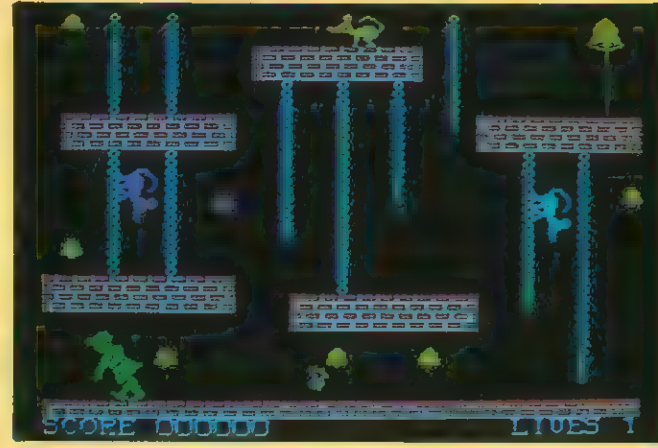
El juego tiene siete pantallas distintas, cada una de las cuales entraña una serie de peligros que tenemos que tratar de evitar ■ toda costa. En las seis primeras, nuestro objetivo es conseguir llegar hasta cada una de las campanas que se encuentran situadas en la parte superior de aquellas. Una vez que hemos obtenido los seis bonos de cada una de las pantallas, podremos acceder a la séptima. En ésta, la dificultad es bastante mayor que en las anteriores, ya que aquí se encuentra Esmeralda, a la cual tenemos que rescatar. Si lo conseguimos volverá a comenzar el juego; pero en esta ocasión, el nivel de dificultad será más alto que en la anterior.

Las pantallas son bastante interesantes desde el punto de vista gráfico, y el movimiento resulta sencillo y práctico para el jugador. Además, es posible utilizar el joystick. Se puede también redefinir las teclas.









El gráfico de Quasimodo es el mismo que en la primera parte; sin embargo, la estructura del programa es, en esta ocasión, mucho más original y las diferentes pantallas están tratadas con mucho más mimo.

Al final de cada juego hay una tabla de records en la cual podemos dejar el nuestro, siempre y cuando logremos alguno.

Valoración: Es divertido y con algunos rasgos de originalidad.

Originalidad	👍👍
Gráficos	👍👍👍
Movimiento	👍👍👍
Valoración	👍👍👍

Las sentencias IN y OUT

LOS «PORTS» DEL SPECTRUM

Jesús ALONSO RODRIGUEZ

Nuestro Spectrum no es un sistema cerrado. Puede comunicarse con otros aparatos y periféricos y, por supuesto, recibir datos y órdenes de los mismos. Todo esto se lleva a cabo a través de los «ports» de entrada y salida, cuyo funcionamiento se analiza en este artículo.

Cuando contemplamos el ordenador desde el punto de vista del programador en Basic, realmente no necesitamos tener una idea muy profunda acerca de su funcionamiento. Trabajamos en base 1Ø, calculamos en coma flotante y utilizamos variables olvidándonos de lo que realmente contienen las posiciones de memoria individuales, o de cómo se las arregla el microprocesador para ejecutar el programa que le hemos introducido.

Pero de pronto, nos encontramos en el teclado algunas sentencias de Basic que no tienen nada que ver con las que utilizamos habitualmente, y en cambio, se refieren al propio Sistema en sí; Comandos como OUT y POKE, y funciones tales como IN, PEEK y USR cuya utilización requiere un mayor conocimiento del Sistema.

En esta ocasión, vamos a abordar el estudio del comando OUT y la función IN, que sirven para comunicarse directamente con los periféricos. Para ello, vamos a estudiar primero, cómo se comunica el microprocesador Z-80 con el exterior, y, a continuación, veremos cómo están ordenados los periféricos dentro del sistema que forma el Spectrum.

Los «ports» del Z-80

El microprocesador Z-80 y la memoria, forman el núcleo central de nuestro ordenador; el microprocesador, se comunica con la memoria a través de un conjunto de «cables» que denominamos los «Buses» del sistema. Existen tres buses: el de direcciones, compuesto por 16 «cables» denominados AØ, A1, A2,... A 15; el de datos, compuesto por 8 «cables» denominados DØ, D1, D2,... D7, y el de control del sistema, compuesto por 6 «cables» denominados «M1», «MREQ», «IORQ», «RD», «WR» y «RFSH». Existen, además, otra serie de «cables» o líneas que controlan el funcionamiento del propio microprocesador y el acceso de otros dispositivos a los buses del sistema, pero en este momento, no nos interesan para lo que pretendemos estudiar.

MREQ significa «Petición de memoria», RD significa «leer» y WR significa «escribir»; si el Z-80 quiere «leer» una posición de memoria, activa las líneas MREQ y RD, y coloca en el bus de direcciones la configuración binaria que corresponde al número de la posición de memoria que desea leer; acto seguido, la memoria le devolverá, a través del bus de datos, la configuración binaria correspondiente al número almacenado en esa posición de memoria. Si lo que deseara fuera escribir un dato, activaría las líneas MREQ y WR colocando la dirección en el bus de direcciones, y el dato a escribir, en el bus de datos.

De forma similar el Z-80 puede leer y escribir datos en los denominados «ports» de entrada/salida». «Port» en inglés significa «puerto», denominación bastante acertada si consideramos que es el lugar donde los datos se mandan al exterior o se reciben del exterior. En el caso de querer acceder a los «ports» el microprocesador activaría la línea IORQ en lugar de la MREQ (IORQ significa «petición de entrada/salida), utilizando de nuevo RD para leer datos y WR para escribirlos.

En este caso, sólo se utiliza la mitad inferior del bus de direcciones, y por tanto, sólo hay 256 «ports» posibles con los que se puede trabajar el Z-80. Cuando el microprocesador desea escribir un dato en un port, activa las líneas IORQ y WR, y coloca en la mitad inferior del bus de direcciones la configuración binaria correspondiente a la dirección del port al que desea acceder, en el bus de datos coloca el dato que desea escribir en ese port y finalmente, en la parte superior del bus de direcciones se coloca el contenido de un registro que puede ser el A o el B dependiendo de la instrucción a la que esté respondiendo el Z-80 en ese momento.



Si lo que desea es leer un dato desde un port, activa las líneas IORQ y RD, y coloca en la mitad inferior del bus de direcciones la configuración binaria correspondiente a la dirección del port del que desea leer; el dato le será entregado desde el port a través del bus de datos; de nuevo, en la mitad superior del bus de direcciones se coloca el contenido de un registro, que podrá ser el A o el B en función de la instrucción a la que esté respondiendo el Z-80 en ese momento. Esta forma de utilizar la mitad superior del bus de direcciones, se revelará muy útil cuando estudiemos la forma en la que se lee el teclado en el Spectrum.

Los periféricos en el Spectrum

En general, se consideran periféricos todo aquello que no sea el microprocesador o la memoria. Aunque el Spectrum se presente como una unidad compacta, de hecho, alberga en su interior, al menos 4 periféricos: teclado, cassette, altavoz y televisor. Excepto el televisor, todos los demás se manejan a través de los ports de entrada/salida. También se manejan mediante los ports todos los restantes periféricos que se pueden incorporar a la unidad básica, tales como joystick, impresora, microdrives, etc.

Hemos mencionado antes que el Z-80 puede direccionar 256 ports distintos, pero con el fin de no complicar la decodificación, el Spectrum utiliza los bits de la mitad inferior del bus de direcciones, de forma independiente; de este modo, todo el control de las comunicaciones se puede delegar en un solo «chip», la famosa «ULA».

En realidad el Spectrum direcciona todos sus periféricos utilizando, solamente, los cinco primero bits del bus de direcciones de una manera bastante ingeniosa, que vamos a intentar explicar. Por tanto, los bits A5, A6 y A7 se encuentran libres para el usuario, o bien para ser utilizados por dispositivos que no sean los de la marca Sinclair (por ejemplo, el interface de joystyck tipo Kempston se maneja con el bit A5 del bus de direcciones).

Para que la ULA identifique que queremos acceder a un periférico determinado, el bit correspondiente ha de ser «cero». Como el ordenador sólo puede hacer una cosa a la vez, sólo uno de los bits ha de valer «cero», mientras los restantes deberán ser «uno».

El bit AØ maneja el teclado y la entrada EAR cuando se direcciona como entrada (lectura) y el altavoz, la salida MIC y el color del borde, cuando se utiliza como salida (escritura). El bit A2 maneja la impresora, y los bits A1 y A4 se utilizan para los microdrives, la ZX-NET y la salida RS-232.

Las direcciones decimales que hacen que estos bits valgan «cero» mientras los demás permanecen a «uno», se muestran en la tabla 1.

El port 254

Posiblemente éste sea el port más importante del Spectrum, ya que si se direcciona como entrada, permite leer el teclado y la entrada EAR, y si se direcciona como salida controla el altavoz, la salida MIC y el color del borde de la pantalla.

El teclado se encuentra dividido en ocho semifilas, todas ellas se leen mediante el port 254, pero la ULA sabe qué semifila leer en función de la configuración binaria que se encuentre presente en la mitad superior del bus de direcciones. Aunque el microprocesador sólo puede direccionar 256 ports, desde el Basic podemos pedirle que nos lea un port con una dirección superior a 255; lo que hará el microprocesador será colocar el octeto inferior de nuestro número (convertido en hexadecimal) en la mitad inferior del bus de direcciones, para determinar a cuál de los 256 ports queremos acceder, y el octeto superior lo colocará en la mitad superior del bus de direcciones para aportar a la ULA la información suplementaria que necesite. Por ejemplo: si le decimos LET a = IN 65Ø22, el microprocesador escribirá 254 en la mitad inferior del bus de direcciones, indicando a la ULA que deseamos leer el teclado, y 253 en la mitad superior, para indicarle que es la semifila que va de la «A» a la «G» la que deseamos leer.

En la Tabla 2 se muestran las ocho direcciones de port que nos permiten leer cada semifila del teclado de forma independiente.

Si al leer una semifila no hay ninguna tecla pulsada, el resultado será 255 (to-

dos los bits del bus de datos a «uno»). Cada tecla que se encuentre pulsada pondrá a cero uno de los cinco primeros bits del bus de datos, DØ para la tecla más exterior, D1 para la siguiente y así sucesivamente hasta D4 para la más cercana a la parte central.

El port 254, configurado como entrada, nos permite también leer el estado de la entrada EAR que actuará en este caso, sobre el séptimo bit del bus de datos, es decir, D7.

Si utilizamos el port 254 como salida, podremos controlar con el bit D4 el altavoz, con D3 la salida MIC y con D2, D1 y DØ, el color del borde de la pantalla. Por ejemplo, si tecleamos el comando OUT 254,3 el borde se pondrá de color magenta; pero recuerde que éste es sólo un color temporal, volverá a su color original cuando pulse cualquier tecla. OUT 254,16 producirá un chasquido en el altavoz mientras el borde se pondrá de color negro.

La función IN

La función IN del Basic nos permite acceder directamente a los ports; su forma general es:

IN dirección

Recuerde que la dirección puede ser superior a 255, por ejemplo, cuando pretenda leer una semifila del teclado.

No olvide que se trata de una función, por lo que teclearla de forma directa como un comando, no tiene sentido. Como toda función, le dará un resultado, que será el valor entregado al bus de datos por el port correspondiente. Este valor deberá asignarlo a una variable o incluirlo en una expresión, por ejemplo: LET a = IN 65Ø22 o bien: IF IN 65Ø22 = 255 THEN GO TO 1ØØ. Encontrará que IN es muy similar a PEEK en cuanto a su sintaxis.

Resulta muy útil la función IN con la dirección 223, ya que nos permite leer el dato entregado por un joystick con interface tipo Kempston.

El comando OUT

Para escribir un dato en un port desde el Basic, hacemos uso del comando OUT; su forma general es:

OUT dirección, dato

En este caso la sintaxis es muy similar a la del comando POKE. Tanto la dirección como el dato deberán ser números comprendidos entre «cero» y 255.

TABLA 1

Bit	Direccion decimal	Configuracion binaria
A0	254	00000000 111 11110
A1	253	00000000 111 11101
A2	251	00000000 111 11011
A3	247	00000000 111 10111
A4	239	00000000 111 01111
A5	223	00000000 110 11111

TABLA 2

Direccion de port	Semifila leida	Configuracion binaria
65278 (FEFE)	CAPS SHIFT a V	11111110 111 11110
65022 (FDFE)	A a G	11111101 111 11110
64510 (FBFE)	Q a T	11111011 111 11110
63486 (F7FE)	1 a 5	11110111 111 11110
61438 (EFFE)	0 a 6	11101111 111 11110
57342 (DFFE)	P a Y	11011111 111 11110
49150 (BFFE)	ENTER a H	10111111 111 11110
32766 (7FFE)	SPACE a B	01111111 111 11110

COMECOCOS

Luis AMADO REGO

Spectrum 48 K

Premiado con 15.000 pts.

Este «Comecocos» es una versión del ya conocido juego que lleva su mismo nombre, en el que hemos de conseguir la máxima puntuación guiando a nuestro «comecocos» por un laberinto.

En este recinto, nuestro amigo ha de comerse el mayor número posible de manzanas (cuyo valor es de 30 puntos) o de semillas de manzana (valoradas en 5 puntos).

Pero no todo va a ser tan placentero para el «comecocos», ya que dos terribles «ogros» le perseguirán de forma encarnizada para eliminarle a lo largo del laberinto. No es fácil eludirles, pero tampoco imposible.

El movimiento se consigue pulsando las teclas 5, 6, 7 y 8 con desplazamientos de acuerdo a lo indicado en las flechas.

NOTAS GRAFICAS

A B C D E F G H I

```
  10 REM        COMECOQUITOS
Luis Amado y MICROHOBBY SEMANAL
  20 GO SUB 680
  30 LET max=0
  40 LET x2=13: LET x3=4: LET y3
=18: LET y2=15: LET px=10: LET p
y=15: LET v$="●": LET j$="◖"
  50 DEF FN p$(a)=("000"+STR$ a)
(LEN STR$ a TO )
  60 BORDER 4: PAPER 6: INK 2: C
LS
  70 LET punt=0
  80 DIM l$(19,30)
  90 LET l$(1)="▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒
▒▒▒▒▒▒▒▒▒▒▒▒▒▒"
 100 LET l$(2)="▒●··············
·············●▒"
 110 LET l$(3)="▒·▒▒▒·▒··▒▒▒·▒▒▒
·▒▒▒··▒·▒▒▒··▒"
 120 LET l$(4)="▒··▒··▒▒▒●▒▒··●·●
··●▒▒▒··▒··▒▒"
 130 LET l$(5)="▒·▒▒·▒▒●▒·▒▒▒▒▒
▒▒▒·▒●▒▒·▒▒·▒▒"
 140 LET l$(6)="▒··▒·····▒·······
····▒·····▒··▒▒"
 150 LET l$(7)="▒·▒▒▒▒·▒▒·▒·▒▒▒▒
▒·▒·▒▒·▒▒▒▒··▒"
 160 LET l$(8)="▒··········▒·▒···
▒·▒···········▒"
 170 LET l$(9)="▒·▒▒▒▒▒▒▒▒▒·▒▒·▒
▒·▒▒▒▒▒▒▒▒▒▒·▒"
 180 LET l$(10)="▒············▒··
·▒·············▒"
 190 LET l$(11)="▒·▒▒▒▒▒▒▒▒▒·▒·▒
·▒·▒▒▒▒▒▒▒▒▒▒·▒"
 200 LET l$(12)="▒·▒·········▒···▒
···▒········●▒·▒"
 210 LET l$(13)="▒·▒▒·▒·▒●▒·▒▒▒▒
▒▒▒··▒·▒·▒·▒▒·▒"
 220 LET l$(14)="▒·▒····▒▒······
····▒·······▒·▒"
 230 LET l$(15)="▒···▒▒▒···▒▒▒▒▒
▒▒▒···▒▒▒▒···▒"
 240 LET l$(16)="▒·▒·····▒▒··▒●·
●▒···▒·······▒·▒"
 250 LET l$(17)="▒·▒▒▒▒▒▒▒▒▒·▒▒▒·
▒▒▒·▒▒▒▒▒▒▒▒▒·▒"
 260 LET l$(18)=l$(2)
 270 LET l$(19)=l$(1)
 280 PRINT : FOR f=1 TO 19: INK
1: PAPER 6: PRINT TAB 1;l$(f): N
EXT f
 290 FOR f=0 TO 21: PRINT INK 0;
AT f,0;"█";AT f,31;"█": NEXT f
 300 PRINT AT 20,0; PAPER 0; INK
7; BRIGHT 1;"      C O M E C O Q
U I T O S     "
 310 PRINT AT 21,0; INK 2; PAPER
6;"PUNTOS:0000           RECORD:
0000"
 320 PRINT AT 0,0; INK 5; PAPER
0;" LAR SOFTWARE  LALIN-PONTEVED
RA "
 330 LET l$(px,py)=" "
 340 IF punt<1745 THEN GO TO 400
 350 IF punt=1745 THEN PRINT AT
2,13; INK 4; PAPER 1; FLASH 1; B
RIGHT 1;"BRAVO";AT 8,14;"HAS";AT
 10,6;"LOGRADO";AT 10,14;"UNA";A
T 10,18;"VICTORIA"
 360 PRINT AT 18,4; INK 7; PAPER
 0; FLASH 1;"QUIERES CONTINUAR?
(s/n)"
 370 IF INKEY$="s" THEN GO TO 60
 380 IF INKEY$="n" THEN GO TO 99
99
 390 IF INKEY$<>"s" OR INKEY$<>"
n" THEN GO TO 370
 400 IF INKEY$="" THEN GO TO 450
 410 IF INKEY$="5" THEN LET j$="
◗"
 420 IF INKEY$="6" THEN LET j$="
◠"
 430 IF INKEY$="7" THEN LET j$="
◡"
 440 IF INKEY$="8" THEN LET j$="
◖"
 450 PRINT AT px,py;" "
 460 IF j$="◡" AND l$(px-1,py)<>
"▒" THEN LET px=px-1
 470 IF j$="◖" AND l$(px,py+1)<>
"▒" THEN LET py=py+1
 480 IF j$="◠" AND l$(px+1,py)<>
"▒" THEN LET px=px+1
 490 IF j$="◗" AND l$(px,py-1)<>
"▒" THEN LET py=py-1
 500 PRINT AT px,py; INK 3;v$
 510 IF l$(px,py)="·" THEN LET p
unt=punt+5: BEEP .01,12
 520 IF l$(px,py)="●" THEN LET p
unt=punt+30: BEEP .02,16
 530 IF max<punt THEN LET max=pu
nt
 540 PRINT AT 21,7;FN p$(punt);A
T 21,28;FN p$(max): BEEP .001,50
 550 PRINT AT x2,y2;l$(x2,y2): I
F INT (RND*2)+(x2>px) AND l$(x2-
1,y2)<>"▒" THEN LET x2=x2-1
 560 IF INT (RND*2)+(x2<px) AND
l$(x2+1,y2)<>"▒" THEN LET x2=x2+
1
 570 IF INT (RND*2)+(y2>py) AND
l$(x2,y2-1)<>"▒" THEN LET y2=y2-
1
 580 IF INT (RND*2)+(y2<py) AND
l$(x2,y2+1)<>"▒" THEN LET y2=y2+
1
 590 PRINT AT x2,y2; INK 2;"☗"
 600 PRINT AT px,py; INK 3;j$
 610 IF (x2=px AND y2=py) OR (x3
=px AND y3=py) THEN FOR g=1 TO 1
0: FOR f=0 TO 7: PRINT INK f;AT
px,py;j$: NEXT f: NEXT g: PRINT
AT px,py; INK 6;j$: GO TO 830
 620 PRINT AT x3,y3;l$(x3,y3): I
F INT (RND*2)+(x3>px) AND l$(x3-
1,y3)<>"▒" THEN LET x3=x3-1
 630 IF INT (RND*2)+(x3<px) AND
l$(x3+1,y3)<>"▒" THEN LET x3=x3+
1
 640 IF INT (RND*2)+(y3>py) AND
l$(x3,y3-1)<>"▒" THEN LET y3=y3-
1
 650 IF INT (RND*2)+(y3<py) AND
l$(x3,y3+1)<>"▒" THEN LET y3=y3+
1
 660 PRINT AT x3,y3; INK 2;"☗"
 670 GO TO 330
 680 DATA 66,129,129,195,231,255
,126,60
 690 DATA 62,121,240,224,224,240
,121,62
 700 DATA 124,158,15,7,7,15,158,
124
 710 DATA 60,126,255,231,195,129
,129,66
 720 DATA 56,124,214,214,254,254
,170,170
 730 DATA 24,82,255,255,255,255,
126,36
 740 DATA 0,60,126,126,126,126,6
0,0
 750 DATA 170,85,170,85,170,85,1
70,85
 760 DATA 0,0,0,24,24,0,0,0
 770 RESTORE 680
 780 FOR i=1 TO 9: FOR n=0 TO 7
 790 READ a
 800 POKE USR CHR$ (i+143)+n,a
 810 NEXT n: NEXT i
 820 RETURN
 830 PRINT AT 2,11; INK 7; PAPER
 1; FLASH 1;"SE  ACABO"
 840 PRINT AT 18,4; INK 7; PAPER
 0; FLASH 1;"QUIERES CONTINUAR?
(S/N)"
 850 IF INKEY$="s" THEN GO TO 40
 860 IF INKEY$="n" THEN GO TO 99
99
 870 IF INKEY$<>"n" OR INKEY$<>"
s" THEN GO TO 850
```



A. PERERA

HIPERBOLOIDE

Javier GUARDIA y Juan TUNEU

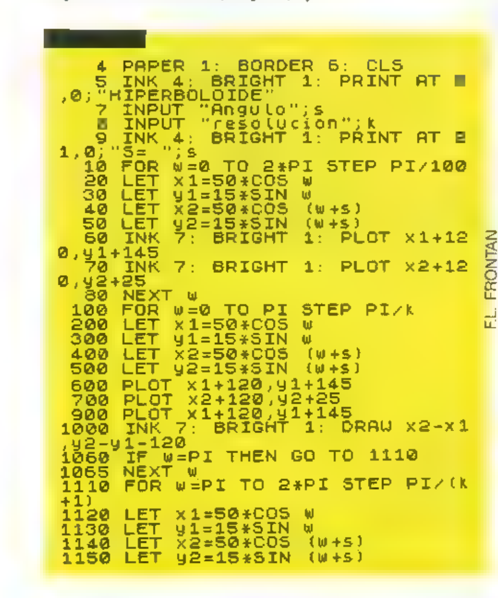
Premiado con 15.000 Ptas.

Spectrum 16 k

Es un programa de utilidades que nos ayudará a la hora de hacer representaciones de figuras hiperbólicas de revolución.

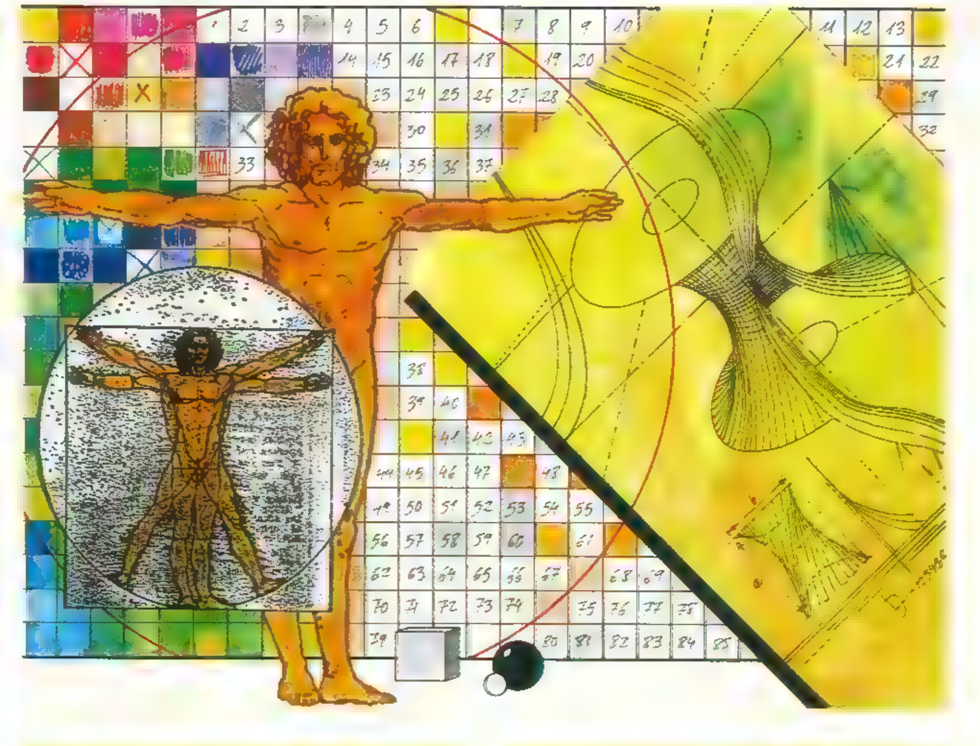
Para su ejecución, puede pedirnos el ordenador el ángulo, para lo que debemos entrar un valor comprendido entre Ø y PI. En este caso debemos probar estos valores extremos con lo que se conseguirá un cilindro y un sistema cónico, respectivamente.

Por otro lado, puede pedirnos también la resolución, en cuyo caso se puede entrar cualquier valor mayor de Ø (aconsejamos entre 1Ø y 5Ø).



```
   4 PAPER 1: BORDER 6: CLS
   5 INK 4: BRIGHT 1: PRINT AT ■
,0;"HIPERBOLOIDE"
   7 INPUT "Angulo";s
   ■ INPUT "resolucion";k
   9 INK 4: BRIGHT 1: PRINT AT 2
1,0;"S= ";s
  10 FOR w=0 TO 2*PI STEP PI/100
  20 LET x1=50*COS w
  30 LET y1=15*SIN w
  40 LET x2=50*COS (w+s)
  50 LET y2=15*SIN (w+s)
  60 INK 7: BRIGHT 1: PLOT x1+12
0,y1+145
  70 INK 7: BRIGHT 1: PLOT x2+12
0,y2+25
  80 NEXT w
 100 FOR w=0 TO PI STEP PI/k
 200 LET x1=50*COS w
 300 LET y1=15*SIN w
 400 LET x2=50*COS (w+s)
 500 LET y2=15*SIN (w+s)
 600 PLOT x1+120,y1+145
 700 PLOT x2+120,y2+25
 900 PLOT x1+120,y1+145
1000 INK 7: BRIGHT 1: DRAW x2-x1
,y2-y1-120
1060 IF w=PI THEN GO TO 1110
1065 NEXT w
1110 FOR w=PI TO 2*PI STEP PI/(k
+1)
1120 LET x1=50*COS w
1130 LET y1=15*SIN w
1140 LET x2=50*COS (w+s)
1150 LET y2=15*SIN (w+s)
```

F.L. FRONTAN



```
1160 PLOT x1+120,y1+145
1170 PLOT x2+120,y2+25
1180 PLOT x1+120,y1+145
1190 INK 7: BRIGHT 1: DRAW x2-x1
,y2-y1-120
1205 IF w=2*PI THEN GO TO 1220
1210 NEXT w
1220 INK 5: IF s=0 THEN PRINT AT
 0,20;"CILINDRO"
1230 INK 5: IF s=PI THEN PRINT A
T 0,20;"SISTEMA CONICO"
1240 INK 5: IF s<>0 AND s<>PI TH
EN PRINT AT 0,20;"HIPERBOLOIDE"
1250 INPUT "REPITES (S/N)";Z$
1260 IF Z$="S" THEN GO TO 0
1270 CLS : PRINT "ADIOS"
9999 SAVE "HIPER" LINE 0: GO TO
9999
```

LAS CUATRO EN RAYA

Ramón BERNAT GARCIA

Premiado con 15.000 pts.

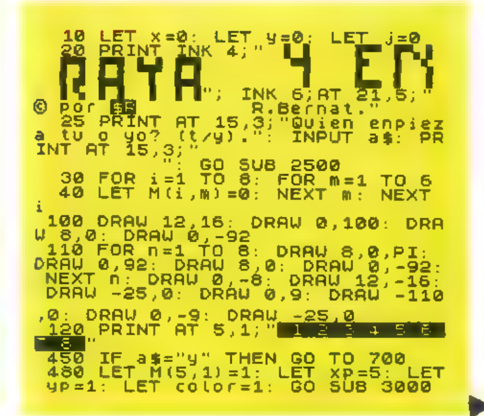
Spectrum 16 K

Las cuatro en raya es un juego muy popular que, sin duda alguna, habremos utilizado en alguna ocasión para pasar un rato entretenido.

Así pues, utilicemos nuestro Spectrum también para estos ratos «perdidos» y dispongámonos a divertirnos un poco teniendo como contrincante a uno muy difícil de superar: el ordenador. El manejo es muy sencillo. Primeramente, aparecerá en pantalla una tabla semejante a las utilizadas en la realidad, en cuya parte superior están definidas y numeradas las columnas. Es en éstas donde tendremos que colocar nuestras fichas, de tal manera que consigamos una recta o diagonal ininterrumpida y del mismo color.

Animate, el reto no es fácil.

```
   5 DIM M(8,6)
   6 BORDER 1: PAPER 1: INK 6: C
LS
```



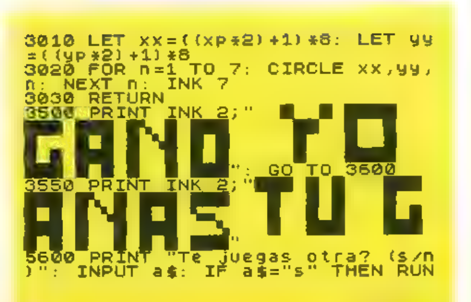
```
  10 LET x=0: LET y=0: LET j=0
  20 PRINT INK 4;"  4 EN
RAYA"; INK 6;AT 21,5;"
© por ■■          R.Bernat."
  25 PRINT AT 15,3;"Quien enpiez
a tu o yo? (t/y).": INPUT a$: PR
INT AT 15,3;"
                 ": GO SUB 2500
  30 FOR i=1 TO 8: FOR m=1 TO 6
  40 LET M(i,m)=0: NEXT m: NEXT
i
 100 DRAW 12,16: DRAW 0,100: DRA
W 8,0: DRAW 0,-92
 110 FOR n=1 TO 8: DRAW 8,0,PI:
DRAW 0,92: DRAW 8,0: DRAW 0,-92:
 NEXT n: DRAW 0,-8: DRAW 12,-16:
 DRAW -25,0: DRAW 0,9: DRAW -110
,0: DRAW 0,-9: DRAW -25,0
 120 PRINT AT 5,1;" 1 2 3 4 5 6
 7 8 "
 450 IF a$="y" THEN GO TO 700
 480 LET M(5,1)=1: LET xp=5: LET
yp=1: LET color=1: GO SUB 3000
```

```
 490 GO TO 700
 500 PRINT AT 12,20; INK 4; FLAS
H 1;"PENSANDO ": LET p=0: LET pp
=0
 510 LET p=0
 520 FOR i=1 TO 8
 530 LET x=i: GO SUB 2000
 540 IF y>6 THEN NEXT i
 550 LET color=1: GO SUB 1030: P
RINT FLASH 1;AT 5,i+20;i
 560 IF psp>p AND psp<30 THEN GO
 SUB 1600
 570 IF psp>p THEN LET p=psp: LE
T xp=x: LET yp=y
 575 IF psp=.05 THEN NEXT i
 580 LET color=2: GO SUB 1030: I
F psp>=30 THEN LET psp=29.9
 590 IF psp>p THEN LET p=psp: LE
T xp=x: LET yp=y
 600 NEXT i: PRINT INK 1; FLASH
0;AT 5,20;"
 610 LET color=1: LET M(xp,yp)=1
 620 GO SUB 2500: GO SUB 3000: L
ET j=j+1
 630 IF p>30 THEN GO TO 3 00
 640 IF j=48 THEN PRINT A 12,20
;"Empate": GO TO 3600
 700 REM ************************
*********  TU JUEGAS ***********
*******************************
 701 PRINT INK 4;AT 12,20;"TU MU
EVES"
 710 INPUT x
 720 IF x<1 OR x>8 THEN GO TO 71
0
 725 LET color=2: GO SUB 2000: I
F y>6 THEN PRINT AT 12,20;"No va
lido": GO TO 710
 730 LET yp=y: LET xp=x: LET col
or=2: GO SUB 3000
 740 LET M(xp,yp)=2: LET j=i+1
 750 REM veo si ganas
 760 GO SUB 1030
 770 IF psp>=30 THEN GO TO 3550
 780 IF j=48 THEN PRINT OVER 1;A
T 21,0;"                ...fin": G
O TO 3600
 790 GO TO 499
1030 LET psp=0: LET np=0
1040 LET dx=1: LET dy=0: GO SUB
1500
1050 LET psp=ps
1060 LET dx=-1: LET dy=0: GO SUB
 1500
1070 LET psp=psp+ps
1080 LET dx=0: LET dy=-1: GO SUB
 1500
1090 IF ps>psp THEN LET psp=ps
1100 LET dx=1: LET dy=-1: GO SUB
 1500
1200 LET np=ps
1210 LET dx=-1: LET dy=1: GO SUB
 1500
1220 LET np=np+ps
1230 IF np>psp THEN LET psp=np
1240 LET dx=1: LET dy=1: GO SUB
1500
1250 LET np=ps
1260 LET dx=-1: LET dy=-1: GO SU
B 1500
1270 LET np=np+ps
1280 IF np>psp THEN LET psp=np
1290 RETURN
1500 LET ps=0: LET xx=x: LET yy=
y: LET b=0
1510 LET xx=xx+dx: LET yy=yy+dy
1520 IF (xx<1) OR (yy<1) OR (xx>
8) OR (yy>6) THEN RETURN
1530 IF M(xx,yy)<>color AND M(xx
,yy)<>0 THEN RETURN
1540 IF M(xx,yy)=color AND b=0 T
HEN LET ps=ps+10: GO TO 1510
1550 LET ps=ps+1: LET b=1: GO TO
 1510
1599 REM ************************
*******************************
```



M. BERROCAL

```
1600 LET M(x,y)=1: LET color=2:
LET y=y+1
1610 GO SUB 1030
1620 IF psp>=30 THEN LET psp=0.0
5
1630 LET y=y-1: LET M(x,y)=0: RE
TURN
2000 LET cont=0
2010 LET cont=cont+1
2020 IF cont>6 THEN LET y=7: RET
URN
2030 IF M(x,cont)<>0 THEN GO TO
2010
2040 LET y=cont: RETURN
2500 FOR n=1 TO 6: BEEP n*n/100,
n: BEEP n/50,n: NEXT n: RETURN
3000 IF color=1 THEN INK 7
3005 IF color=2 THEN INK 2
```



```
3010 LET xx=((xp*2)+1)*8: LET yy
=((yp*2)+1)*8
3020 FOR n=1 TO 7: CIRCLE xx,yy,
n: NEXT n: INK 7
3030 RETURN
3500 PRINT INK 2;"GANO YO": GO TO 3600
3550 PRINT INK 2;"GANAS TU G"
5600 PRINT "Te juegas otra? (s/n
)": INPUT a$: IF a$="s" THEN RUN
```

ORDENALETRAS

Premiado con 15.000 Ptas.

Juan C. RIBES BENET

Spectrum 48 K

Con este juego demostrarás tu rapidez a la hora de ordenar alfabéticamente una serie de letras que aparecerán, aleatoriamente, en un gran tablero.

Este es, en síntesis, el objetivo de este programa en el que podrás elegir entre tres tamaños diferentes de tablero:

— 3 × 3, con 8 letras.
— 4 × 4, con 15 letras.
— 5 × 5. con 24 letras.

NOTAS GRAFICAS

A ▌ B ▀

Inicialmente, el tablero aparece ordenado, y se desordena aleatoriamente siguiendo el mismo proceso de ordenación. Así se evita el que se plantee una ordenación imposible.

A continuación, se efectúa la ordenación pasando las letras a una cuadrícula en blanco, existente en los tres tipos de tablero, quedando otro espacio en blanco en el lugar de la letra que hemos cambiado. Siguiendo este método, y las instrucciones que vienen insertadas en el propio programa, se ha de ordenar alfabéticamente todo el tablero utilizando para ello las siguientes teclas del cursor: 5, 6, 7 y 8.

```
   1 BORDER 1: PAPER 6: INK 0: C
LS : GO TO 3
   2 FOR g=1 TO 75: LET f=INT (R
ND*22): LET s=INT (RND*32): LET
t=INT (RND*8): LET cd=INT (RND*2
6)+65: PRINT AT f,s; INK t;CHR$
cd: NEXT g: PAUSE 25: PRINT ; IN
K 1;AT 8,12;"▛▀▀▀▀▀▀▜";AT 9,12;"
▌ MESA ▐";AT 10,12;"▌  DE  ▐";AT
 11,12;"▌LETRAS▐";AT 12,12;"▙▄▄▄
▄▄▄▟": INK 0: PAUSE 200: CLS
   3 GO SUB 375
   ▪ CLS : LET rec=10↑6: LET NUM
=0
  10 FOR i=0 TO 7: POKE USR "a"+
i,126: NEXT i: POKE USR "b",0: P
OKE USR "b"+1,0
  15 FOR i=1 TO 6: POKE USR "b"+
i,255: NEXT i
  20 LET a$="ABCDEFGHIJKLMNOPQRS
TUVXY"
  25 INPUT "NUMERO DE LETRAS 8,1
5,24 ? ";n
  30 IF n=8 THEN GO TO 50
  35 IF n=1▪ THEN GO TO 60
  40 IF n=24 THEN GO TO 70
  45 GO TO 25
  50 LET pr1=102: LET pr2=113: L
ET d=43: LET ghyi=9: LET ghyf=11
: LET ghxi=13: LET ghxf=17: LET
gvxi=14: LET gvxf=16: LET gvyi=8
: LET gvyf=12: LET lsx=17: LET l
ix=13: LET lsy=12: LET liy=8
  55 GO TO 75
  60 LET pr1=94: LET pr2=121: LE
T d=59: LET ghyi=8: LET ghyf=12:
 LET ghxi=12: LET ghxf=18: LET g
vxi=13: LET gvxf=17: LET gvyi=7:
 LET gvyf=13: LET lsx=18: LET li
x=12: LET lsy=13: LET liy=7
  65 GO TO 75
  70 LET pr1=86: LET pr2=129: LE
T d=75: LET ghyi=7: LET ghyf=13:
 LET ghxi=11: LET ghxf=19: LET g
vxi=12: LET gvxf=18: LET gvyi=6:
 LET gvyf=14: LET lsx=19: LET li
x=11: LET lsy=14: LET liy=6
  75 INK 2: PLOT pr1,pr2: DRAW d
,0: DRAW 0,-d: DRAW -d,0: DRAW 0
,d: DRAW -1,1: DRAW d+2,0: DRAW
0,-(d+2): DRAW -(d+2),0: DRAW 0,
d+2
  80 FOR f=ghyi TO ghyf STEP 2:
FOR c=ghxi TO ghxf: PRINT INK 4;
AT f,c;"▀": NEXT c: NEXT f
  85 FOR c=gvxi TO gvxf STEP 2:
FOR f=gvyi TO gvyf: PRINT OVER 1
; INK 4;AT f,c;"▌": NEXT f: NEXT
 c: INK 0
  90 PRINT AT 2,7;"ESPERA UN MOM
ENTO"
  95 LET spx=lsx: LET spxn=spx:
LET spy=lsy: LET spyn=spy
 100 LET b$=a$(1 TO n): LET k=0
 105 FOR y=liy TO lsy STEP 2
 110 FOR x=lix TO lsx STEP 2
 115 LET k=k+1
 120 PRINT AT y,x;b$(k TO k)
 125 IF k=n THEN GO SUB 135
 130 NEXT x: NEXT y
 135 GO SUB 315
 140 PRINT AT 2,4;"NUMERO DE INT
ENTOS= ";num
 145 PAUSE 0: IF CODE INKEY$<53
OR CODE INKEY$>56 THEN GO TO 145
 150 IF INKEY$="7" THEN LET spyn
=spy+2
 155 PRINT AT 2,24;num
 160 IF INKEY$="6" THEN LET spyn
=spy-2
 165 IF INKEY$="5" THEN LET spxn
=spx+2
 170 IF INKEY$="8" THEN LET spxn
=spx-2
 175 IF spxn>lsx THEN LET spxn=l
sx: GO TO 145
 180 IF spxn<lix THEN LET spxn=l
ix: GO TO 145
 185 IF spyn>lsy THEN LET spyn=l
sy: GO TO 145
 190 IF spyn<liy THEN LET spyn=l
iy: GO TO 145
 195 LET num=num+1
 20▪ PRINT AT 2,24;num
 205 GO SUB 220
 210 GO SUB 245
 215 GO TO 145
 220 LET m$=SCREEN$ (spyn,spxn)
 225 PRINT AT spy,spx;m$
 230 PRINT AT spyn,spxn;"_"
 235 LET spx=spxn: LET spy=spyn
 240 RETURN
 245 LET p$="A"
 250 FOR o=liy TO lsy STEP 2
 255 FOR a=lix TO lsx STEP 2
 260 LET s$=SCREEN$ (o,a)
 265 IF s$<p$ THEN RETURN
 270 LET p$=s$
 275 NEXT a: NEXT o
 280 PRINT AT 17,7;"LO HAS CONSE
GUIDO"
 285 IF num<rec THEN LET rec=num
: PRINT AT 19,7;"NUEVO RECORD: "
;REC: GO TO 295
 290 PRINT AT 19,13;"RECORD: ";r
ec
 295 POKE 23658,8: PRINT AT 4,1;
 FLASH 1;"QUIERES SEGUIR JUGANDO
? (S/N)"
 300 PAUSE 0: IF INKEY$="N" THEN
 GO TO 310
 305 CLS : GO TO 10
 310 CLS : PRINT AT 9,0;"    EST
AS SEGURO DE NO SEGUIR      JUG
ANDO?  (S/N)": PAUSE 0: IF INKEY
$<>"S" THEN GO TO 305
 311 IF INKEY$="S" THEN CLS : PR
INT AT 10,5;"BUENO,PUES ADIOS!":
 FOR f=9 TO 0 STEP -1: PRINT AT
10,25;f: PAUSE 10: NEXT f: RANDO
MIZE USR 0
 312 GO TO 310
 315 LET q=INT (RND*3)+3: FOR i=
1 TO q*n
 320 LET rev=INT (RND*2)
 325 IF rev=0 THEN LET spyn=spyn
+2: IF spyn>lsy THEN LET spyn=ls
y: GO TO 320
 330 IF rev=1 THEN LET spyn=spyn
-2: IF spyn<liy THEN LET spyn=li
y: GO TO 320
 335 GO SUB 220
 340 LET rev=INT (RND*2)
 345 IF rev=0 THEN LET spxn=spxn
+2: IF spxn>lsx THEN LET spxn=ls
x: GO TO 340
 350 IF rev=1 THEN LET spxn=spxn
-2: IF spxn<lix THEN LET spxn=li
x: GO TO 340
 355 GO SUB 220
 360 NEXT i
 365 PRINT AT 17,13;"JUEGA!"
 370 RETURN
 375 PRINT AT 1,2;"ESTO ES MUY F
ACIL:";AT 2,2;"TIENES QUE ORDENA
▪ LAS LETRAS  DE UN TABLERO,EN E
L MINIMO NU-  MERO DE INTENTOS P
OSIBLE.";AT 6,1;"PARA ORDENAR,SO
LO PUEDES MOVER  LAS LETRAS,CAMB
IANDOLAS POR EL  ESPACIO EN BLAN
CO.";AT 10,1;"UTILIZA LAS TECLAS
 DEL CURSOR.   LA DIRECCION QUE
PULSES  SERA  A LA QUE SE  MOVER
A  L▪ LETRA  PARA OCUPAR EL ESP
ACIO LIBRE.";AT 15,1;"PUEDES ELE
GIR 3 DIMENSIONES DE  TABLERO."
 380 PRINT AT 18,1;"AL COMIENZO
APARECE EL TABLERO  ORDENADO Y
ALEATORIAMENTE  SE  DESORDENA."
 385 PRINT #1; PAPER 3; BRIGHT 1
; FLASH 1;AT 1,8;"PULSA UNA TECL
A": PAUSE 0
 390 RETURN
```

El galardonado en el sorteo mensual correspondiente a los Programas de Lectores, recibidos en enero, ha sido JAVIER LOPES ESCAMEZ, que vive en la calle GRANADA, 64, 6.° A. 04003 ALMERIA.

¡¡¡ENHORABUENA!!!



GRAFICOS EN MOVIMIENTO (III)

Miguel SEPULVEDA

La rutina de movimiento más esperada por los aficionados es, sin duda, la que desplaza los gráficos pixel a pixel por la pantalla. En esta tercera parte de la serie abordamos cómo hacerlo.

En primer lugar, había que decir que en esta rutina el movimiento de las figuras por la pantalla es lento si lo comparamos con la del artículo anterior, que va saltando de 8 en 8 pixel; pero la suavidad con que mueve esta rutina la figura, sin saltos bruscos, es asombrosa si tenemos en cuenta que se mueve de pixel en pixel.

El tamaño de las figuras que puede manejar puede ir, desde un simple pixel, hasta una figura que tenga 255 pixel (32 octetos) por scan y una altura de 176 scan; como se puede comprobar, maneja desde un simple pixel hasta un figura que ocupe toda la pantalla y, en este caso, se puede usar para hacer el scroll de la pantalla hacia arriba, pixel a pixel.

También esta rutina puede mover las figuras suavemente, o sea pixel a pixel, o más rápidamente, moviéndola de dos en dos pixel o de tres en tres, etc. dependiendo del incremento que se haga en las coordenadas X e Y de la esquina superior del gráfico al llamar a la rutina, que sitúa la figura en pantalla a partir de esas dos coordenadas.

Visto un poco las grandes posibilidades que tiene esta rutina, pasamos a explicar el programa que acompaña a este artículo.

Las 1Ø sentencias DATA contienen los octetos del código máquina de la rutina que se carga en la posición de memoria 64ØØØ, y tiene una longitud de 238 octetos. La línea 8Ø hace una comprobación de la suma de los octetos por si al introducir el programa se produce un error.

La sentencia de la línea 7Ø almacena la rutina en la posición de memoria 64ØØØ; pero como esta rutina es reubicable cambiando los valores del bucle FOR... NEXT y, teniendo en cuenta la longitud de la rutina que es de 238 octetos, se puede incluir en la parte de memoria que se quiera, siempre que sea por encima de la RAMTOP, que está situada en la dirección de memoria 27999.

Lo primero que nos pide la rutina son las figuras creadas con el programa del primer artículo, ya que es precisamente a éstas, a las que dotamos de movimiento. A continuación, pide el número de bits (pixel) que tiene de anchura el scan (no tiene por qué ser necesariamente el número de octetos del scan multiplicado por 8), y, luego, pide el número de scan.

Más adelante nos requerirá los atributos con los que queremos que se muestre la figura en la pantalla, y hace una demostración moviendo la figura diagonalmente 8 veces. Para terminar, el programa preguntará si se quiere salvar la rutina en cinta de cassette, teniendo la precaución de cambiar la dirección del SAVE « » CODE si se ha ubicado la rutina en otro lugar de la memoria.

Las variables usadas por esta rutina, también van en el buffer de la impresora en las direcciones siguientes:

- 233ØØ — Número de la coordenada X de la esquina superior izquierda de la figura.
- 233Ø1 — Número de la coordenada Y de la esquina superior izquierda de la figura.
- 233Ø2 — Es el MODO. 1 muestra figura. Ø borra figura.
- 233Ø3 — Número de bits (pixel), o sea la longitud del scan en bits.
- 233Ø4 — Número de scan de la figura.
- 233Ø5 — Los atributos con los que se muestra la figura.
- 233Ø6 — Octeto bajo la dirección de la figura en memoria.
- 233Ø7 — Octeto alto de la dirección de la figura en memoria.

Además esta rutina utiliza, desde la 233Ø8, hasta la posición 23314 para su uso interno.

La variable MODO, con el valor 1 coge los atributos de la parte de la pantalla donde se muestra la figura y los guarda, pone los nuevos atributos y muestra la figura.

Si la variable MODO es Ø, entonces borra la figura y restituye los atributos que tenía anteriormente.

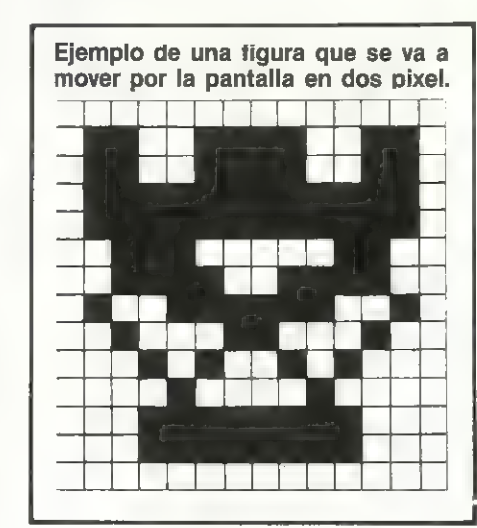
Como se verá en la demostración de esta rutina, muestra la figura y, a continuación, la borra, lo que quiere decir que hay que llamar 2 veces a la rutina, la primera vez con la variable MODO igual a 1, y la segunda con la variable MODO igual a Ø.

De este modo la velocidad de la figura por la pantalla es lenta, pero se puede aumentar el doble si sólo se muestra la figura y no se borra, en cuyo caso irá dejando una estela detrás de ella de los últimos pixel de la figura, en la dirección que se mueva.



A. PERERA

Ejemplo de una figura que se va a mover por la pantalla en dos pixel.



Para evitar esto, hay un truco que consiste en dejar tantos pixel en blanco alrededor de la figura, al crearla, como el incremento que se haga de la coordenada correspondiente en el movimiento de la figura.

En esta demostración se incrementan las coordenadas en cada pasada tantos pixel como el número de la pasada, o sea, de uno en un pixel, de dos en dos pixel, etc, hasta 8.

En la variable 233Ø5 va el octeto de los atributos de la figura que se forma como en el BASIC (ver figura 2).

7	6	5	4	3	2	1	0	BIT
FLASH	BRIGHT	PAPER			INK			

El listado 2 corresponde a los nemotécnicos de la rutina para facilitar su modificación en caso de interés y adaptarla a su propio juego.

PROGRAMA CARGADOR

```
  10 DATA 243,221,33,4,91,237,75,4,91
  12 DATA 205,170,34,50,14,91,221,52,10,1,0,1,221,134,3,61,56,5,12,214,8,24,249,237,67
  14 DATA 12,91,93,124,15,15,15,230,3,246,88,87,213,237,91,10,91,235,58,8,91,8,213,58,7
  16 DATA 91,50,16,91,58,14,91,50,15,91,221,203,2,70,32,8,6,0,221,54,13,250,24,6,70
  18 DATA 35,221,54,13,8,14,8,26,221,53,11,40,7,7,13,221,53,11,32,249,203,32,23,221,53
  20 DATA 12,40,23,221,53,13,32,6,70,35,221,54,13,8,13,32,234,18,19,26,14,8,24,227,24
  22 DATA 176,13,40,4,7,13,32,252,18,209,8,254,1,40,49,8,229,235,124,230,7,254,7,40,3
  24 DATA 36,24,30,125,230,224,254,224,40,11,221,52,9,17,224,6,167,237,82,24,12,124,254,87,40
  26 DATA 7,221,52,9,17,32,0,25,235,225,8,61,32,191,225,17,32,0,221,203,2,70,32,5,58
  28 DATA 18,91,24,7,126,50,18,91,58,9,91,221,78,9,221,70,8,229,119,35,16,252,225,25,13,32,243,251,201
  60 CLEAR 27999: LET c=0
  70 FOR n=64000 TO 64237: READ a: LET c=c+a: POKE n,a: NEXT n
  80 IF c<>20312 THEN PRINT "ERROR DE CHECKSUM": STOP
 100 INPUT "QUIERE CARGAR LA FIGURA "; LINE a$
 110 IF a$="S" OR a$="s" THEN LOAD ""CODE
 130 INPUT "Num. de bits de anchura= ";nb: POKE 23303,nb
 140 INPUT "Num. de scan de altura= ";ns: POKE 23304,ns
 150 CLS : PRINT AT 2,5;"ATRIBUTOS DE LA FIGURA";AT 4,1;"Dar el numero del PAPEL y la  TINTA y 1 para activar el FLASH y BRIGHT y 0 para desactivarlos": LET at=0
 160 INPUT "PAPER= ";a: LET at=a*8
 170 INPUT "INK= ";a: LET at=at+a
 180 INPUT "BRIGHT= ";a: LET at=at+a*64
 190 INPUT "FLASH= ";a: LET at=at+128*a: POKE 23305,at
 200 CLS : INPUT "Dar octetos bajo y alto de la  direc. de la figura= ";a;" - ";b: POKE 23306,a: POKE 23307,b
 205 LET a=1
 210 FOR f=1 TO 8
 220 FOR n=175 TO ns STEP -a
 230 POKE 23302,1: POKE 23300,n: POKE 23301,n: RANDOMIZE USR 64000
 240 POKE 23302,0: RANDOMIZE USR 64000
 250 NEXT n
 255 LET a=a+1
 260 NEXT f
 270 INPUT "QUIERE SALVAR LA RUTINA "; LINE a$
 280 IF a$="S" OR a$="s" THEN SAVE "Dibpixels"CODE 64000,238
 290 STOP
```

DESENSAMBLE DE LA RUTINA

```
FA00          0010       ORG  64000
5B04          0020 CORDX EQU  23300
5B04          0030 COLUM EQU  23300
5B05          0040 CORDY EQU  23301
5B05          0050 LINEA EQU  23301
5B06          0060 MODO  EQU  23302
5B07          0070 ANCHO EQU  23303
5B07          0080 NBYTE EQU  23303
5B08          0090 ALTO  EQU  23304
5B08          0100 NSCAN EQU  23304
5B09          0110 ATTR  EQU  23305
5B0A          0120 FIGUR EQU  23306
5B0C          0130 NATTL EQU  23308
5B0D          0140 NATTH EQU  23309
5B0E          0150 REST  EQU  23310
5B0F          0160 RESS  EQU  23311
5B10          0170 ANCH  EQU  23312
5B11          0180 BUCLE EQU  23313
5B12          0190 ATTRO EQU  23314
              0200 ;
              0210 ;           PROGRAMA
              0220 ;
FA00          0230 DIBUJ EQU  $
FA00 F3       0240       DI
FA01 DD21045B 0250       LD   IX,CORDX
FA05 ED4B045B 0260       LD   BC,(CORDX)
FA09 CDAA22   0270       CALL 22AAH
FA0C 320E5B   0280       LD   (REST),A
FA0F DD340A   0290       INC  (IX+10)
FA12 010001   0300       LD   BC,100H
FA15 DD8603   0310       ADD  (IX+3)
FA18 3D       0320       DEC  A
FA19 3805     0330 DB0   JR   C,DB2
FA1B 0C       0340       INC  C
FA1C D608     0350       SUB  8
FA1E 18F9     0360       JR   DB0
FA20 ED430C5B 0370 DB2   LD   (NATTL),BC
FA24 5D       0380       LD   E,L
FA25 7A       0390       LD   A,D
FA26 0F       0400       RRCA
FA27 0F       0410       RRCA
FA28 0F       0420       RRCA
FA29 E603     0430       AND  3
FA2B F658     0440       OR   58H
FA2D 57       0450       LD   D,A
FA2E D5       0460       PUSH DE
FA2F ED5B0A5B 0470       LD   DE,(FIGUR)
FA33 EB       0480       EX   DE,HL
FA34 3A085B   0490       LD   A,(ALTO)
FA37          0500 DIB0  EQU  $
FA37 08       0510       EX   AF,A'F'
FA38 D5       0520       PUSH DE
FA39 3A075B   0530       LD   A,(ANCHO)
FA3C 32105B   0540       LD   (ANCH),A
FA3F 3A0E5B   0550       LD   A,(REST)
FA42 320F5B   0560       LD   (RESS),A
FA45 DDCB0246 0570       BIT  0,(IX+2)
FA49 200B     0580       JR   NZ,DIB2
FA4B 0600     0590       LD   B,0
FA4D DD360DFA 0600       LD   (IX+13),250
FA4D DD360DFA 0600       LD   (IX+13),250
FA51 1806     0610       JR   DIB6
FA53          0620 DIB2  EQU  $
FA53 46       0630       LD   B,(HL)
FA54 23       0640       INC  HL
FA55          0650 DIB4  EQU  $
FA55 DD360D08 0660       LD   (IX+13),8
FA59          0670 DIB6  EQU  $
FA59 0E08     0680       LD   C,8
FA5B 1A       0690       LD   A,(DE)
FA5C DD350B   0700       DEC  (IX+11)
FA5F 2807     0710       JR   Z,DIB10
FA61          0720 DIB8  EQU  $
FA61 07       0730       RLCA
FA62 [illegible] 0740    DEC  C
FA63 DD350B   0750       DEC  (IX+11)
FA66 20F9     0760       JR   NZ,DIB8
FA68          0770 DIB10 EQU  $
FA68 CB20     0780       SLA  B
FA6A 17       0790       RLA
FA6B DD350C   0800       DEC  (IX+12)
FA6E 2817     0810       JR   Z,DIB14
FA70 DD350D   0820       DEC  (IX+13)
FA73 2006     0830       JR   NZ,DIB12
FA75 46       0840       LD   B,(HL)
FA76 23       0850       INC  HL
FA77 DD360D08 0860       LD   (IX+13),8
FA7B          0870 DIB12 EQU  $
FA7B 0D       0880       DEC  C
FA7C 20EA     0890       JR   NZ,DIB10
FA7E 12       0900       LD   (DE),A
FA7F 13       0910       INC  DE
FA80 1A       0920       LD   A,(DE)
FA81 0E08     0930       LD   C,8
FA83 18E3     0940       JR   DIB10
FA85          0950 DIB13 EQU  $
FA85 18B0     0960       JR   DIB0
FA87          0970 DIB14 EQU  $
FA87 0D       0980       DEC  C
FA88 2804     0990       JR   Z,DIB18
FA8A          1000 DIB16 EQU  $
FA8A 07       1010       RLCA
FA8B 0D       1020       DEC  C
FA8C 20FC     1030       JR   NZ,DIB16
FA8E          1040 DIB18 EQU  $
FA8E 12       1050       LD   (DE),A
FA8F D1       1060       POP  DE
FA90 08       1070       EX   AF,A'F'
FA91 FE01     1080       CP   1
FA93 2831     1090       JR   Z,DIB26
FA95 08       1100       EX   AF,A'F'
FA96 E5       1110       PUSH HL
FA97 EB       1120       EX   DE,HL
FA98 7C       1130       LD   A,H
FA99 E607     1140       AND  7
FA9B FE07     1150       CP   7
FA9D 2803     1160       JR   Z,DIB20
FA9F 24       1170       INC  H
FAA0 181E     1180       JR   DIB24
FAA2          1190 DIB20 EQU  $
FAA2 7D       1200       LD   A,L
FAA3 E6E0     1210       AND  0E0H
FAA5 FEE0     1220       CP   0E0H
FAA7 280B     1230       JR   Z,DIB22
FAA9 DD3409   1240       INC  (IX+9)
FAAC 11E006   1250       LD   DE,6E0H
FAAF A7       1260       AND  A
FAB0 ED52     1270       SBC  HL,DE
FAB2 180C     1280       JR   DIB24
FAB4          1290 DIB22 EQU  $
FAB4 7C       1300       LD   A,H
FAB5 FE57     1310       CP   57H
FAB7 2807     1320       JR   Z,DIB24
FAB9 DD3409   1330       INC  (IX+9)
FABC 112000   1340       LD   DE,20H
FABF 19       1350       ADD  HL,DE
FAC0          1360 DIB24 EQU  $
FAC0 EB       1370       EX   DE,HL
FAC1 E1       1380       POP  HL
FAC2 08       1390       EX   AF,A'F'
FAC3 3D       1400       DEC  A
FAC4 20BF     1410       JR   NZ,DIB13
FAC6          1420 DIB26 EQU  $
FAC6 E1       1430       POP  HL
FAC7 112000   1440       LD   DE,32
FACA DDCB0246 1450       BIT  0,(IX+2)
FACE 2005     1460       JR   NZ,DIB28
FAD0 3A125B   1470       LD   A,(ATTRO)
FAD3 1807     1480       JR   DIB30
FAD5          1490 DIB28 EQU  $
FAD5 7E       1500       LD   A,(HL)
FAD6 32125B   1510       LD   (ATTRO),A
FAD9 3A095B   1520       LD   A,(ATTR)
FADC          1530 DIB30 EQU  $
FADC DD4E09   1540       LD   C,(IX+9)
FADF          1550 DIB32 EQU  $
FADF DD4608   1560       LD   B,(IX+8)
FAE2 E5       1570       PUSH HL
FAE3          1580 DIB34 EQU  $
FAE3 77       1590       LD   (HL),A
FAE4 23       1600       INC  HL
FAE5 10FC     1610       DJNZ DIB34
FAE7 E1       1620       POP  HL
FAE8 19       1630       ADD  HL,DE
FAE9 0D       1640       DEC  C
FAEA 20F3     1650       JR   NZ,DIB32
FAEC FB       1660       EI
FAED C9       1670       RET
```

CONSULTORIO

Spectrum y Spectrum Plus

Quiero comprarme un ordenador y quisiera saber cuál me aconsejáis vosotros; el ZX Spectrum 48 K o el Spectrum Plus.

Unos familiares míos me aconsejan el primeramente citado, por que opinan que es el que mejor satisfaría mis necesidades, pero yo pienso que tiene poca memoria y el segundo me convendría más, por tener mejor teclado y más memoria.

José A. LORO-Madrid

□ En la elección de un ordenador influyen un gran número de factores y nosotros no podemos aconsejarle un determinado modelo sin conocer sus necesidades, no obstante le podemos informar de que los dos modelos que nos menciona tienen la misma capacidad de memoria, de hecho se trata de la misma máquina y solo se diferencian en el teclado.

Cassete para el QL

Deseo saber qué posibilidades hay para acoplar un cassette al QL, o si está previsto en un futuro aplicarle algún dispositivo de cassette.

Antonio A. RIVAS-Oviedo

□ No tenemos noticias de que exista ningún sistema para acoplar un cassette al QL. Realmente no es necesario, puesto que los Microdrives cumplen la función ventajosamente.

Deterioro del cassette

En una revista leí que con el uso continuado de los programas llega a borrarse de la cinta la información que contiene, de ser esto cierto, ¿cuánto tiempo dura la información en la cinta?

José F. FULLERA-Valencia

□ Efectivamente, las cintas de cassette se deterioran con el paso continuado de los cabezales a través de ellas.

No es posible determinar con exactitud la duración de una cinta, ya que eso depende de sus características, que no son iguales para todas; no obstante, no debe preocuparse por ello, ya que lo más posible es que se canse usted del programa antes de que la cinta empiece a fallar.

Carga de programas

Al poner en el ordenador la sentencia RANDOMIZE USR 1234 sale el signo de salvarse algo, ¿qué es?; ¿por qué sale sin cabecera?

¿Podríais hacer una rutina para aumentar la velocidad de grabación?

¿Cuál es la instrucción en código máquina para cargar programas?

¿Cómo puedo parar un programa BASIC con autoejecución?

Rafael DE MINGO-Madrid

□ La sentencia RANDOMIZE USR 1234 obliga al microprocesador a saltar a la posición de memoria Ø4C2. Esta posición de memoria cae en medio de la rutina de salvar bytes que se encuentra en la ROM y que empieza en la posición Ø4D2, es decir, 16 bytes antes.

No aconsejamos a nuestros lectores que hagan esta prueba, ya que el entrar en la rutina por este punto desordena la pila de máquina por lo que no será posible el retorno al Basic y el ordenador se quedará bloqueado.

La rutina para aumentar la velocidad de grabación la tiene en los números 11, 12 y 13 de nuestra revista.

No existe ninguna instrucción en código máquina que sirva para cargar programas, el intérprete de basic lo hace mediante una serie de rutinas compuestas por varias decenas de instrucciones.

Cualquier programa en Basic se puede parar pulsando simultáneamente las teclas de «espacio» y «mayúsculas» siempre y cuando no esté protegido. Si desea que un programa grabado con auto-ejecución no se auto-ejecute, cárguelo con "MERGE" en lugar de "LOAD".

Cassette sin contador

Me gustaría saber la forma de separar los programas grabados en cinta, de forma que al querer usar alguno de ellos los encuentre sin problemas. Os planteo este problema porque el cassette de que dispongo no tiene contador.

Juan J. GARCIA-Cádiz

□ Por razones obvias la cinta de cassette no admite formateado, por lo que la única manera de localizar rápidamente un programa es utilizando el contador del cassette; no obstante, si Vd. teclea LOAD y a continuación el nombre del programa, entre comillas, el ordenador ignorará todos los programas que se encuentre antes con distinto nombre.

Algoritmo de ordenación

Soy poseedor de un Spectrum de 48 K y desearía poder desarrollar en él un programa de estas características:

Se introducen unos nombres que el Spectrum se encarga de ordenar según el alfabeto. Ruego me indiquen el listado con los cambios oportunos.

J. J. ROMERO-Navarra

□ No es misión de este apartado desarrollar rutinas «de encargo»; no obstante como nuestra intención es satisfacer a todos los lectores, le indicamos el procedimiento a seguir:

1. Cargue las palabras a ordenar en una matriz de caracteres.
2. Utilice un bucle que, empezando por el primer elemento de la matriz, vaya comparando los contiguos dos a dos e intercambiándolos si el segundo ha de ser anterior al primero.
3. Para realizar la comparación, recuerde que el Spectrum es capaz de ordenar series de cadenas por orden alfabético, ya que considerará «menor» la cadena que alfabéticamente deba ir antes.

Este método se conoce como «Algoritmo de burbuja»; en la cinta de demostración encontrará un ejemplo.

Ampliación de memoria

Tengo un ZX Spectrum de 16 K que me gustaría ampliar a 48 K comprándome una memoria externa de 32 K, pero si más tarde me comprara los dos interfaces y el microdrive, ¿podría conectar la ampliación externa de 32 K con los interfaces o con la impresora en el Spectrum?

Eduardo GILABERTE-Madrid

□ Los periféricos que nos indica son perfectamente compatibles y podrá conectarlos simultáneamente sin problema.

Frankenstein Informático

Quiero adquirir un ordenador, pero mientras no lo tenga no puedo utilizar el cartucho de 10 juegos II, entonces quisiera saber si se puede conectar un radiocassette a los video-juegos ATARI y poner el cartucho de 10 juegos II, y si se puede qué hay que hacer.

Sergio ALVAREZ-Gerona

□ Lo que Vd. pretende hacer es construir un «monstruo» informático. Por razones obvias jamás funcionaría.

Auto-ejecución

1. ¿Cómo se hace para que un programa empiece a funcionar sin necesidad de RUN?

2. ¿Cómo se consigue que un muñeco, nave, etc. al moverse no deje rastro?

3. ¿Cómo se puede escribir un programa en Cobol en el Spectrum?

Rodrigo FERNANDEZ-Madrid

□ Para que un programa empiece a funcionar sin necesidad de RUN, grábelo de la siguiente forma: SAVE "nombre" LINE 1

Para conseguir que un muñeco no deje rastro al moverse, imprima primero un espacio en la posición ocupada por el muñeco, a continuación incremente o decremente la variable que corresponda al movimiento que deba realizar y finalmente vuelva a imprimir el muñeco en la nueva posición.

No tenemos noticia de que exista ningún compilador de Cobol para el Spectrum.

Almacenamiento de pantallas

¿Es posible recuperar una pantalla después de ser cargada en el ordenador con LOAD ""SCREEN$ y haberla borrado?

David ARTRUGA-Valladolid

□ Como es lógico, una pantalla no se puede recuperar después de haberla borrado, lo que sí puede hacer es transferirla previamente a otra zona de memoria. Si a continuación la borra, siempre podrá volverla a transferir al archivo de pantalla.

Esta transferencia, en Basic, lleva varios segundos; pero en código máquina puede realizarlo en menos de medio segundo haciendo uso de las instrucciones LDIR o bien LDDR.

Reset

Desearía saber si es más perjudicial quitar un programa por un interruptor on/off o bien por el reset. ¿El montaje del reset del número 1 es el mismo que lleva montado el Spectrum Plus? Y por último, en caso de poner un on/off, ¿dónde es mejor ponerlo, en el cable que va de la red al transformador o en el cable que va del transformador al ordenador?

Alvaro AGUIRRE-Madrid

□ Es indiferente borrar un programa con el reset o con el interruptor, pero en este segundo caso, asegúrese de dejar transcurrir al menos diez segundos entre la desconexión y la conexión.

El reset aparecido en nuestro número 1 es exactamente el mismo que incorpora el Spectrum Plus.

El interruptor podrá colocarlo en cualquiera de los dos cables propuestos.

Error en listado

He realizado el pograma ATERRIZA COMO PUEDAS del número 4, y al correrlo me indica error en la línea 112Ø. Como mis conocimientos sobre programación son todavía de «primaria», no he sido capaz de analizar el error, el cual humildemente sospecho que es de impresión de la revista, por lo cual les agradecería me confirmaran la bondad del mismo.

Antoni GARCIA-Barcelona

□ Le confirmamos que el listado aparecido en la revista funciona correctamente. Con mucho gusto intentaríamos analizar su error si es tan amable de indicarnos qué mensaje da exactamente su ordenador.

Evitar cortes de luz

Quisiera saber si existe algún acumulador comercial con entrada a 220 v. salida a 9 v. y amperaje suficiente para alimentar mi Spectrum y evitar el fastidioso problema de los cortes de luz.

Fernando MEIJIDE-Toledo

□ No tenemos noticias de que exista ningún acumulador comercial con las características que usted nos solicita, pero puede dedicarse al «bricolage» y hacérselo utilizando cualquier acumulador capaz de suministrar 2 amperios durante un tiempo razonable.

Avería

Querría preguntarles por qué al conectar el Spectrum en vez de salir el típico mensaje de Sinclair aparecen unas bandas verticales y gruesas sobre las cuales bailan unos cuadraditos de colores. También quiero saber para qué sirve un interruptor que existe detrás de la fuente de alimentación.

Joaquín TIZON-Madrid

□ Sentimos comunicarle que lo más posible es que su ordenador se encuentre averiado; la avería puede estar localizada en el Bus de direcciones del Z-80 o bien en los circuitos TTL que se utilizan para direccionar la memoria. Este tipo de averías se producen con frecuencia por enchufar los interfaces en el Slot con el ordenador conectado, o bien por tocar el Slot con los dedos mientras el ordenador está funcionando.

El interruptor que se encuentra en la parte posterior de la fuente de alimentación sirve para conectar y desconectar el ordenador.

(cortar por la línea de trazos)

Franqueo
Postal

HOBBY PRESS, S. A.

Apartado de Correos

n.° **54.062** (Apartados Altos)

MADRID

DE OCASION

• COMPRO Spectrum 48 K. En buen estado. Completo, con manual en castellano. Juegos y revistas sobre el tema. Preguntar por Federico. Tlf. 70 17 57 , horas de comida. Alicante.

• VENDO para Spectrum, sintetizador de voz Currah Microspeech (agosto 84). Haz hablar a tu Spectrum. Sonido por altavoz TV. Prácticamente sin usar. Manual y cinta demostración en inglés. 6.700 pts. contra reembolso. Interesados escribir a Javier Rovira. C/ Palau Reial, 12, 2.º 07001 Palma de Mallorca (Baleares).

• VENDO ZX 81 completo (manual, cables...) ampliado a 16 K. Libro de código máquina. Tres cintas con programas. Precio: 10.000 pts. Llamar a Juan José Rivero Aranda. Tlf. 455 68 09. C/ Orense, 32 5.º C. Madrid. D.N.I. 2527171.

• VENDO ZX 81 comprado en abril de 1984. Como nuevo. Incluye manual, transformador y todo lo necesario para su uso y además una cassette con cinco juegos. Precio: 13.000 pts. Interesados dirigirse a: Alberto Fernández Cachero. C/ Gijón, 19, 3.º A. Piedras Blancas (Asturias), o bien llamando al teléfono (985) 53 29 02, a partir de las seis y media de la tarde.

• DESEO adquirir el cable que une el radio-cassette con el ordenador, que posee las clavijas de EAR y MIC. Estaría dipuesto a dar mil pts. o más (negociable). Interesados escribir a: Vía paisajista B.L. Alcor 5.º. Huelva D.P. 21003, o llamar al: (21) 24 34 15, de Huelva. Por favor piénsenselo que són mil pelas. Preguntar por Chano.

• VENDO computador Chess Champion, exclusivo de ajedrez, dificultad progresiva, varios niveles, usado tan sólo dos veces, con transformador y conector especial para red. 15.000 pts. Escribir o llamar a: José Tarroja Martínez. C/ Villar, 40, 2.º 1.ª - 08026 Barcelona. Tlf. (93) 347 91 14.

• DESEARIA contactar con usuarios del ZX Spectrum, para intercambiar programas e ideas, interesados, escribir a Gaspar Gómez Rodríguez. C/ Argos, 35. Tlf. 43. Vva. de los Castillejos (Huelva).

• VENDO ZX Spectrum de 16 K, nuevo, con garantía. Más la cinta de «Horizontes», el alimentador, el manual de Basic y el de instrucciones en castellano, los cables y conexiones. Todo por sólo 25.000 pts. M.ª del Mar Arnaez. C/ Empecinado 3. 47003 Valladolid. Tlf. (983) 25 16 85.

• CAMBIO: Ordenador ZX-Spectrum 48 K completamente nuevo más 10.000 pts. por un Commodore 64. Si le interesa, escriba a: Roger Mayola Castillo. C/ Dos de Mayo 327.08025. Barna. Tlf. 235 82 94.

• ESTOY interesado en comprar una impresora ZX de ocasión. Escribir a Agustí Castells. C/ Servet, 106. bajos. 08030 Barcelona.

• VENDO o cambio Vídeo-juego ATARI con su respectiva consola, transformador de corriente, dos clases de mandos de paleta y joystick con cuatro cartuchos de juegos, todo nuevo y en perfecto estado por sólo 18.000 pts., o cambio por periféricos para Spectrum 48 K como impresora Sheikoska (abonando la diferencia), o por cintas de juegos y utilidades por el valor del mismo. Escribir a Raúl Domingo Jordi. Urb. Sant Joan. C/ Cinca, 7. Reus (Tarragona).

• VENDO Spectrum 64 K (Plus), comprado hace 9 días, con garantía INVESTRONICA y con siete cassettes de juegos para Spectrum: Manic Miner, Bandera a cuadros, Jet Pac, etc. Angel Casas. Tlf. (93) 205 02 64.

• VENDO Sinclair XZ 81, nuevo, precio discutible. Regalo libro BASIC sinclair. Carlos Tont Sánchez. Elche (Alicante). Tlf. 46 00 79, llamar de 8 a 10 h.

• ME interesaría contactar con usuarios del SPECTRUM que tengan programas originales, a ser posible en la provincia o ciudad de Barcelona, aunque no importa que sean del resto de España. Carlos Pedra Durán. C/ Pompeu Fabra, nº 42 ATC. Sta. Coloma de Gramanet. Tfn.: 386 06 45.

• VENDO vídeo-juegos ATARI con dos juegos de mandos, varios cartuchos e instrucciones para su manejo. Están en buenísimo estado. Dirigirse al teléfono: 28 46 69. José Ramón Silveira Rodriguez. La Coruña.

RESERVE SU EJEMPLAR

¡EN EL PROXIMO NUMERO REPETIMOS LA CASSETTE!

Atención al próximo número de «MICROHOBBY Semanal», porque, **al mismo precio de siempre, 95 pesetas,** repetimos la edición de la cinta con el programa «Hobby Suerte». No queremos que nadie se quede sin ella, para que todos puedan seguir, semana a semana, las incidencias del concurso.

Reserve ya en su kiosco un ejemplar del número próximo con una cassette, **¡completamente gratis !**